

Kafka DevOps: Maintenance through Monitoring



Justin Pihony

DEVELOPER SUPPORT MANAGER @ LIGHTBEND

@JustinPihony



Maintenance



Monitoring Basics

Key Metrics

Demo

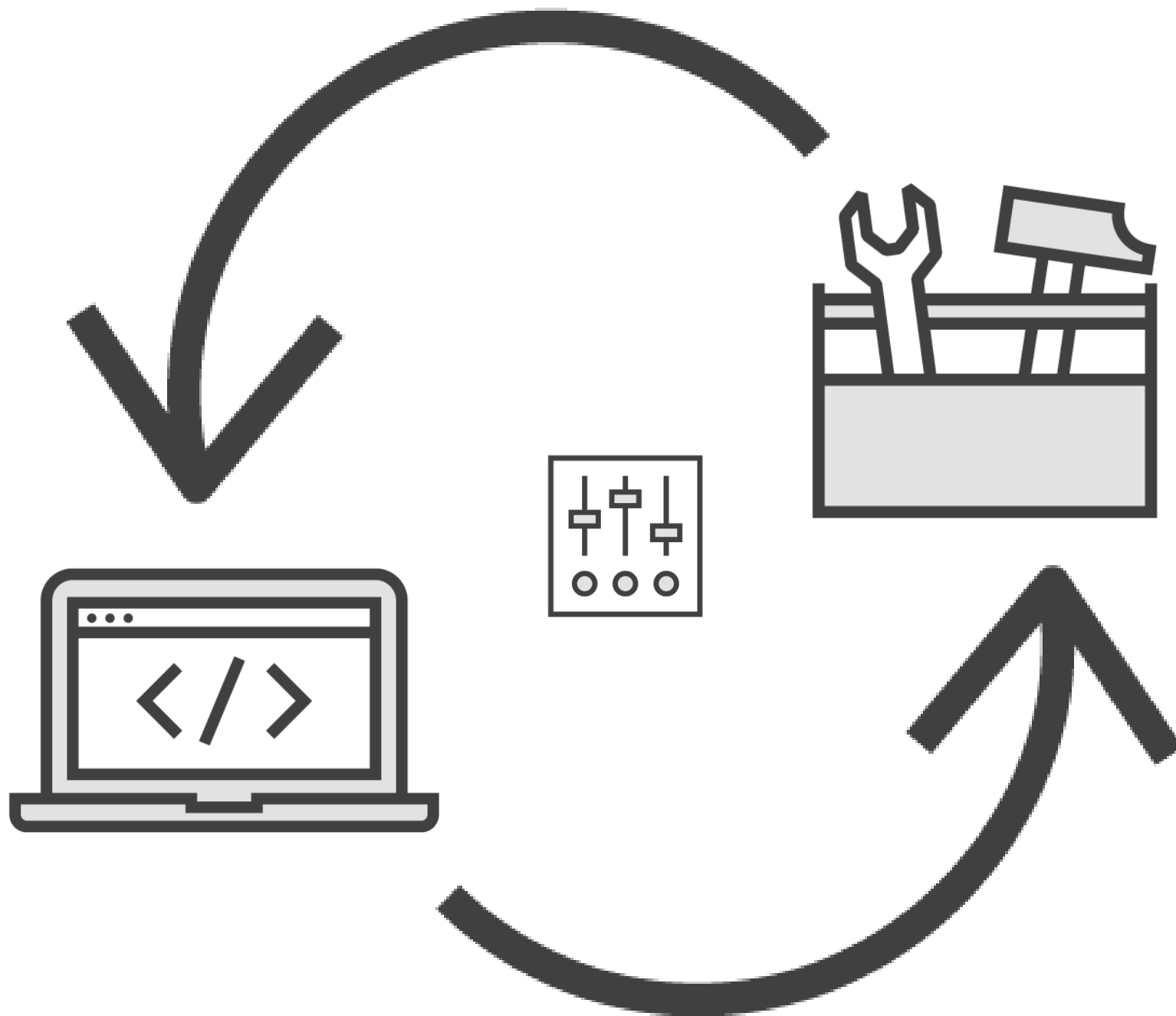
Maintaining is More than Metrics

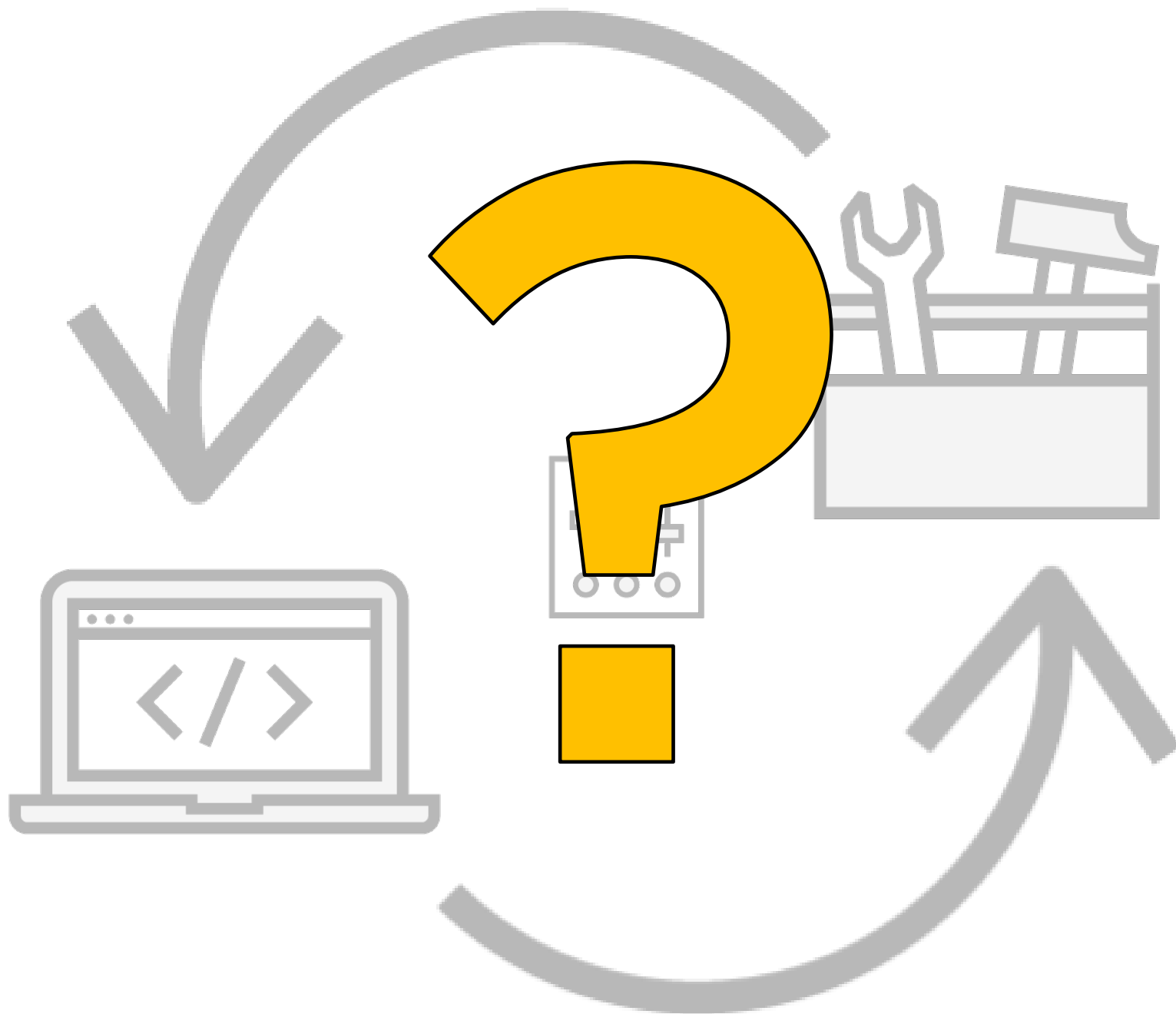


Monitoring Basics





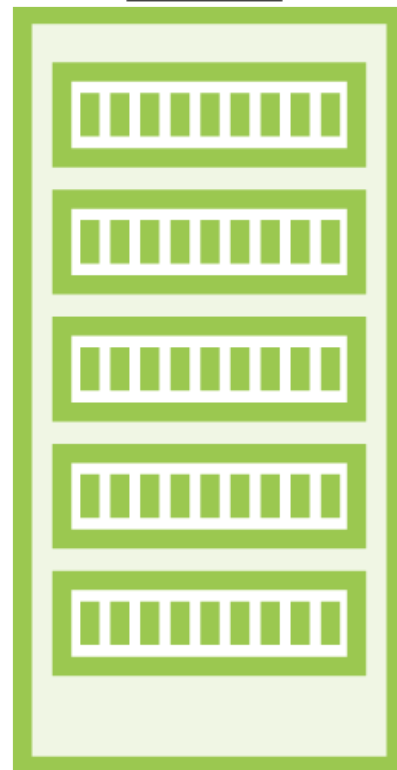
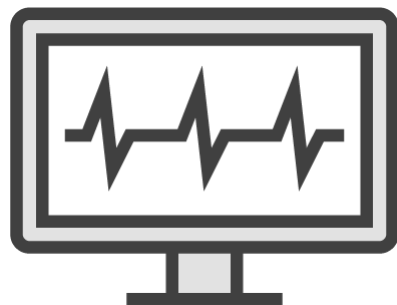


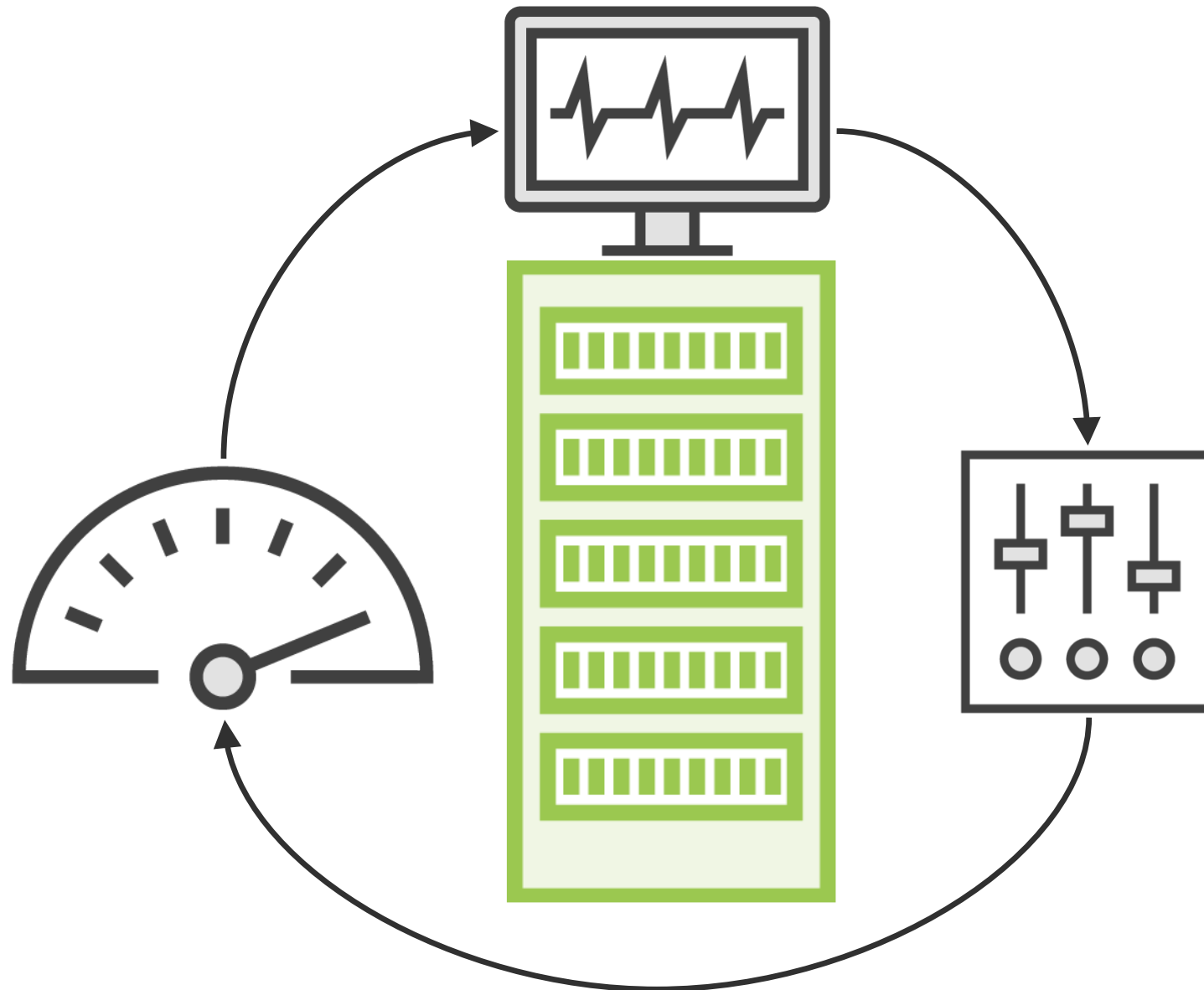


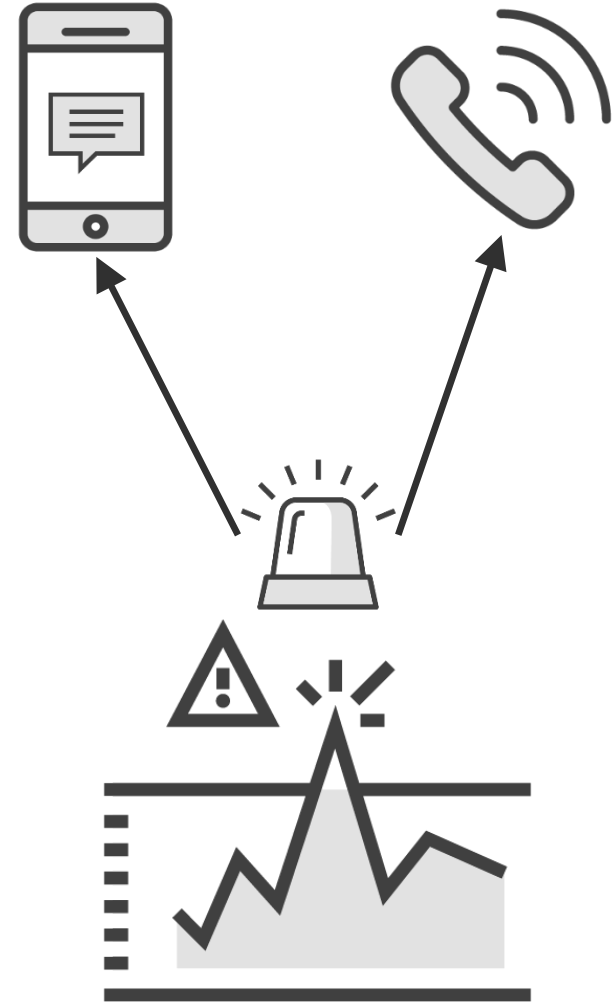




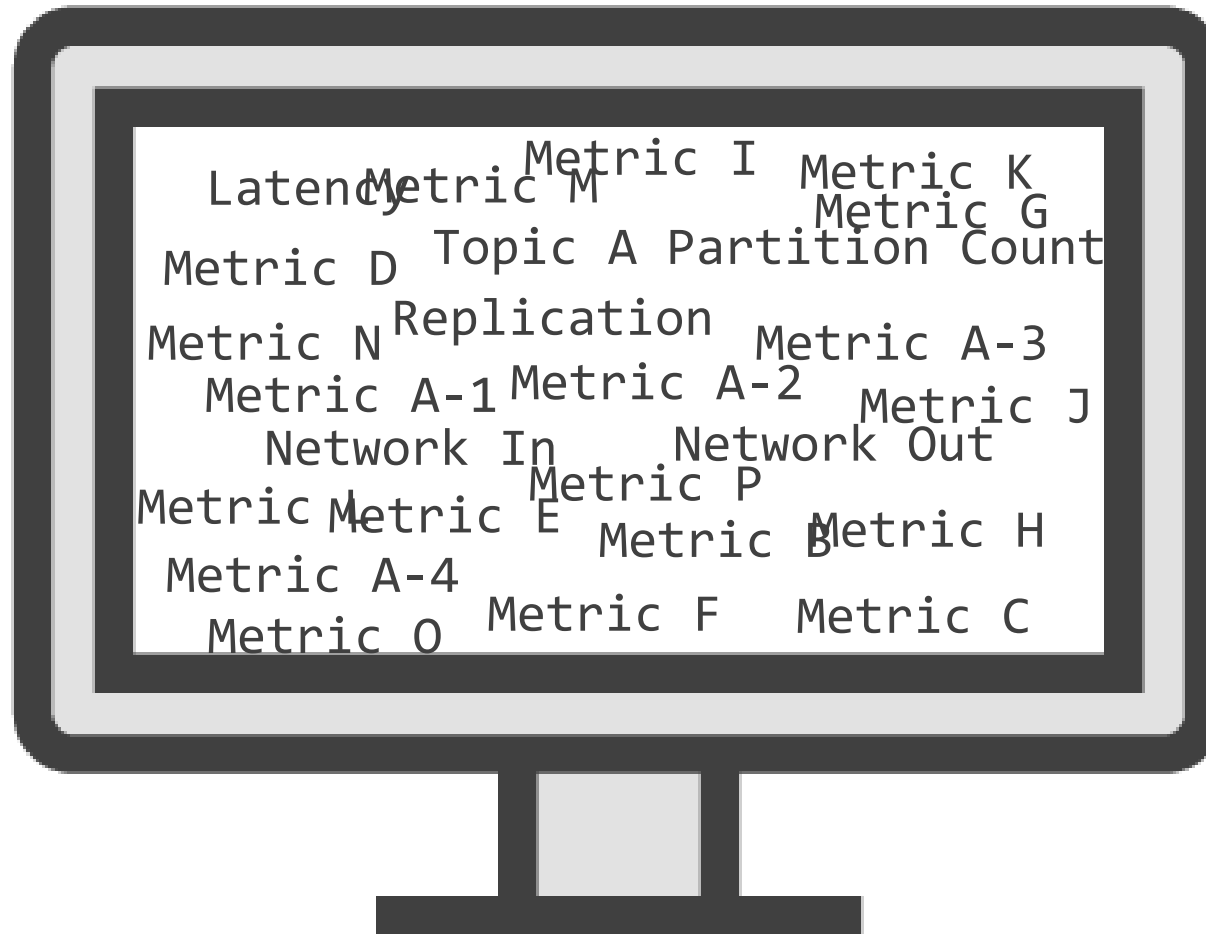








Kafka Metrics Galore!



Kafka Metrics Galore!



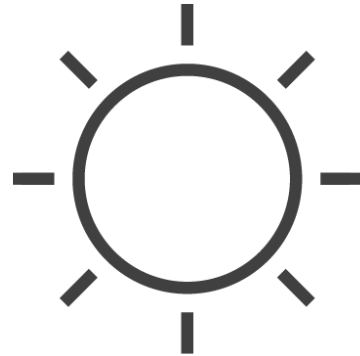
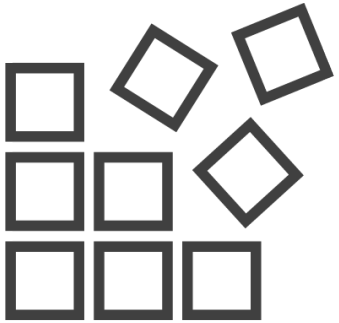
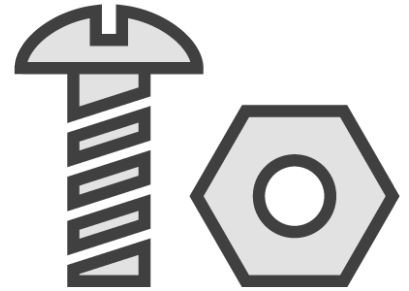
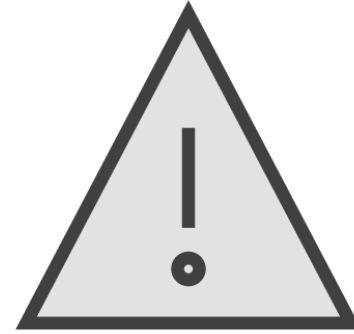
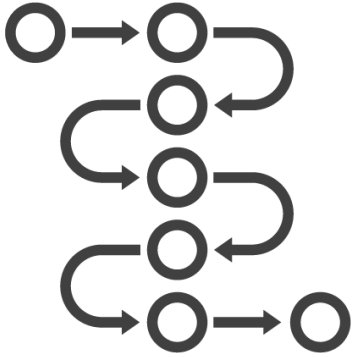
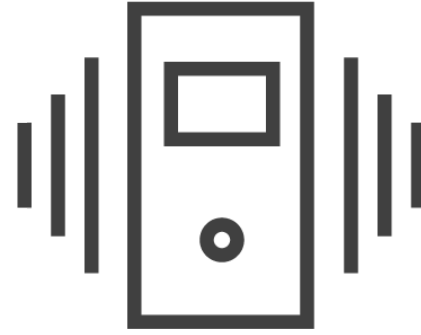
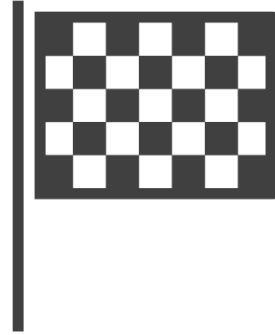
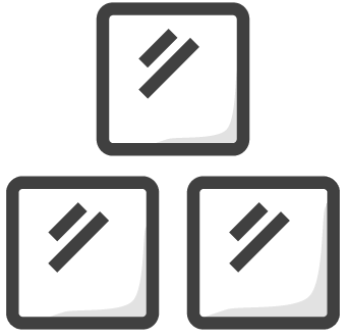
Kafka Metrics Galore!



Kafka Metrics Galore!





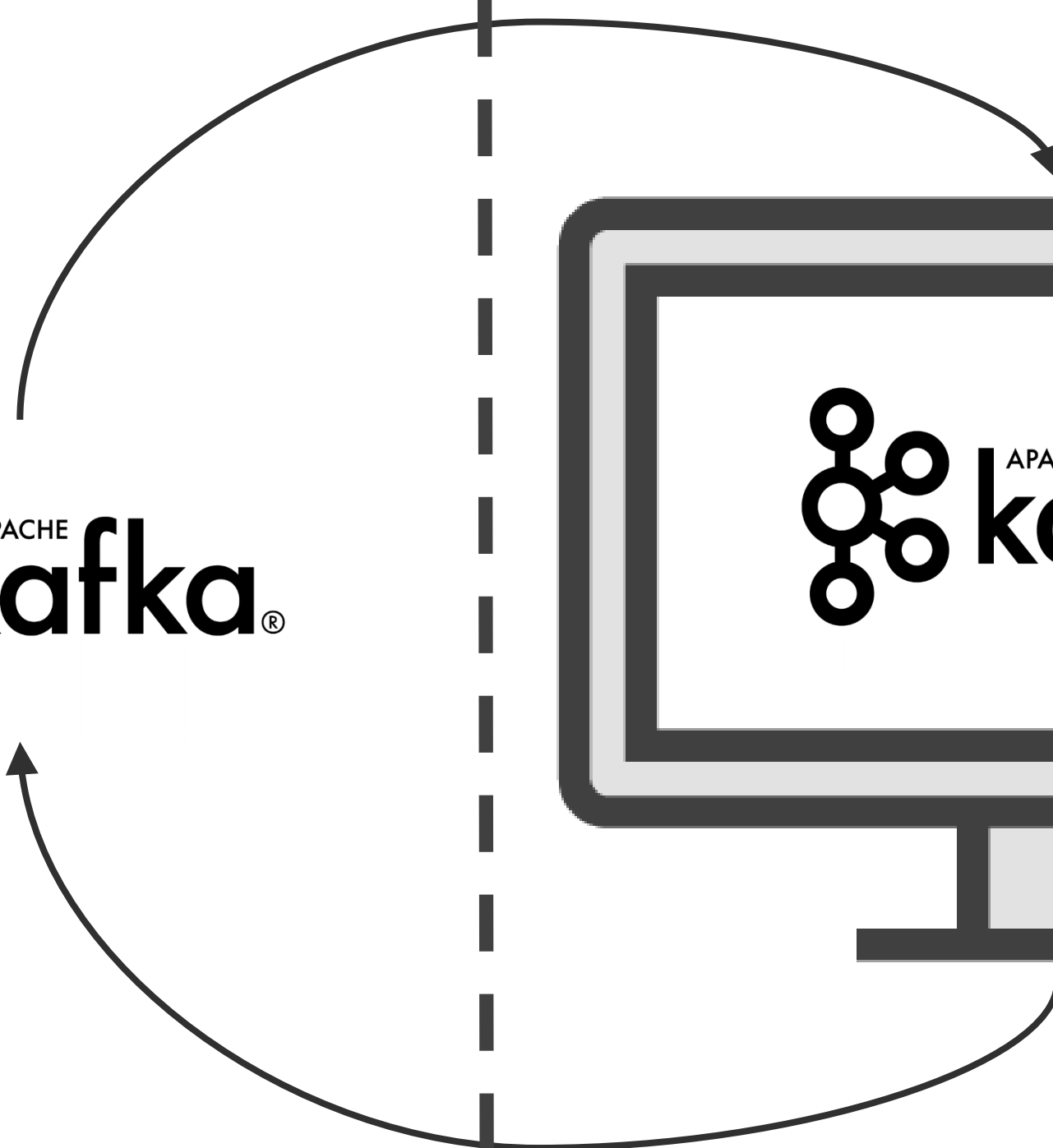


Focused Kafka Metrics

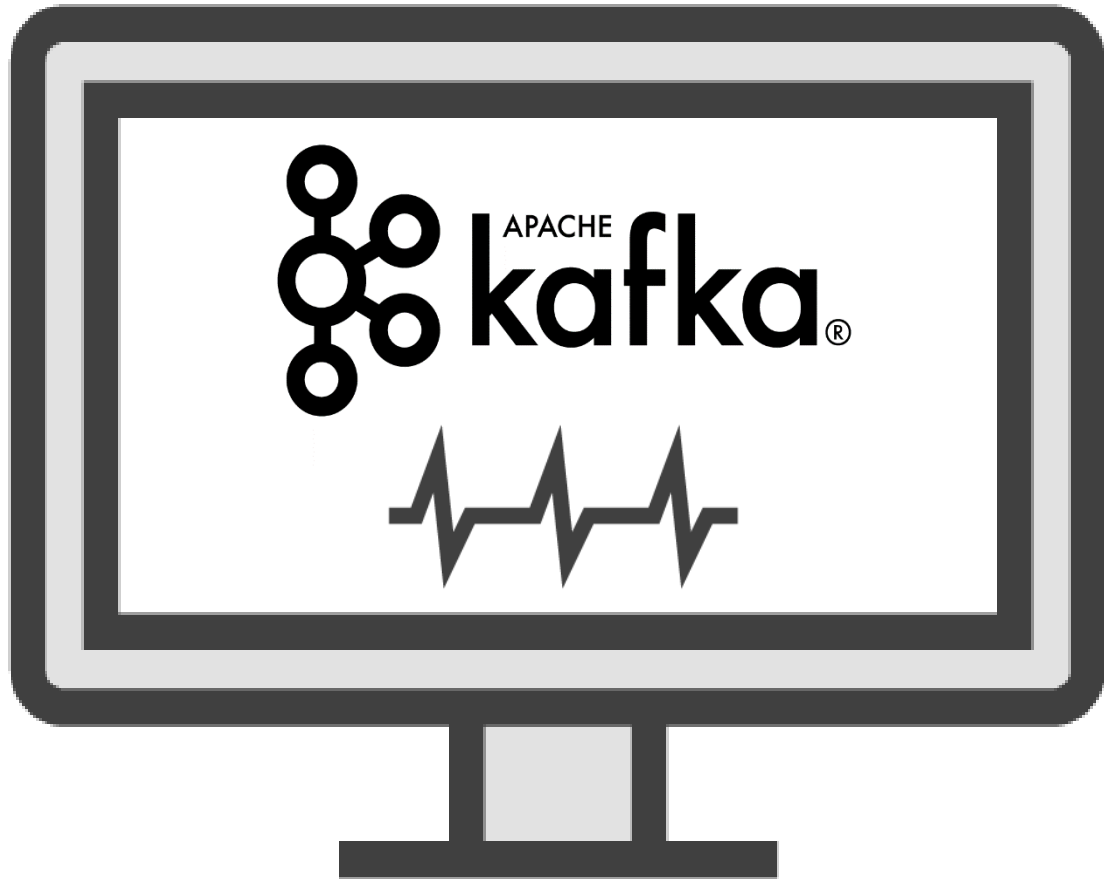








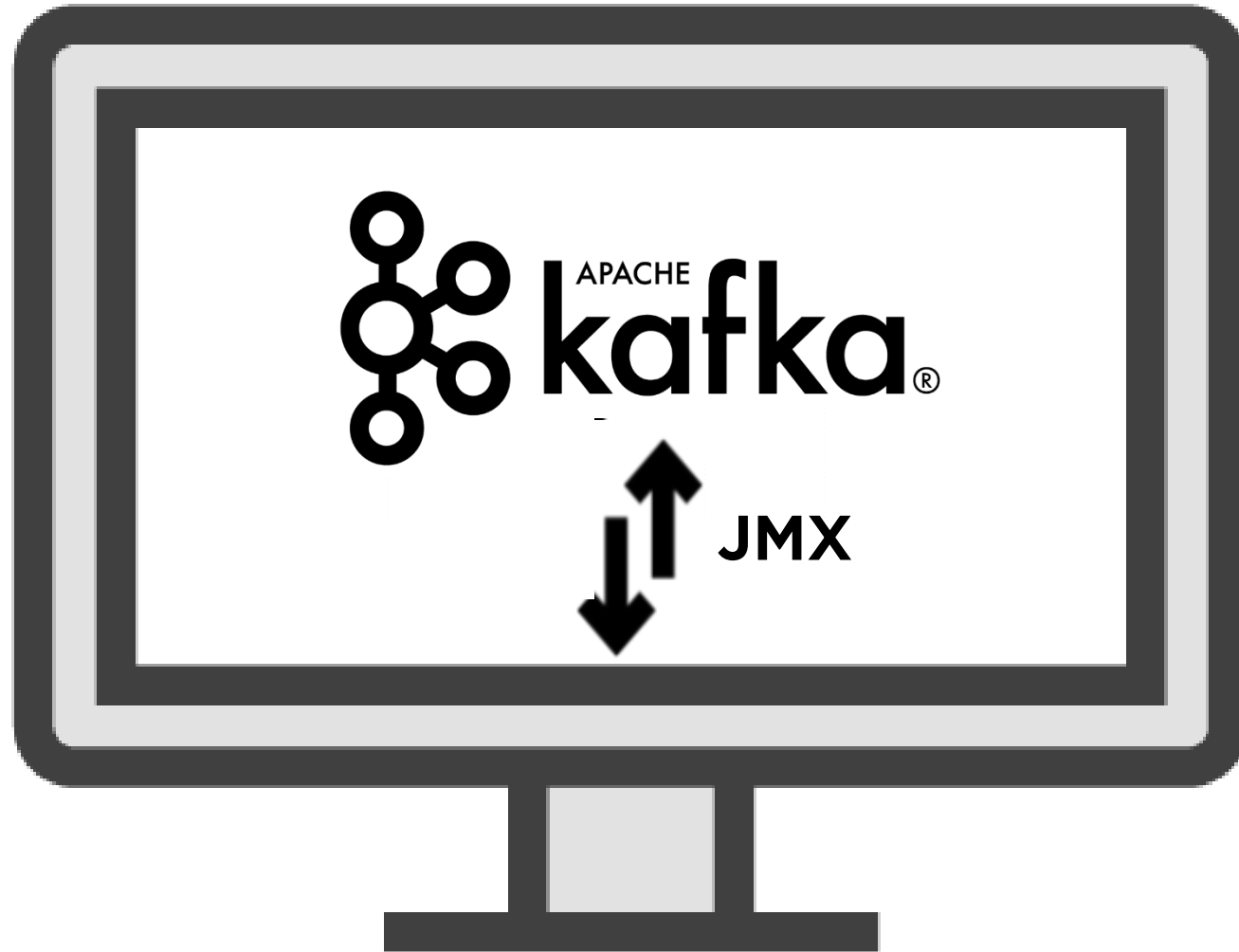
Internal / External



Key Metrics: Under-replicated Partitions

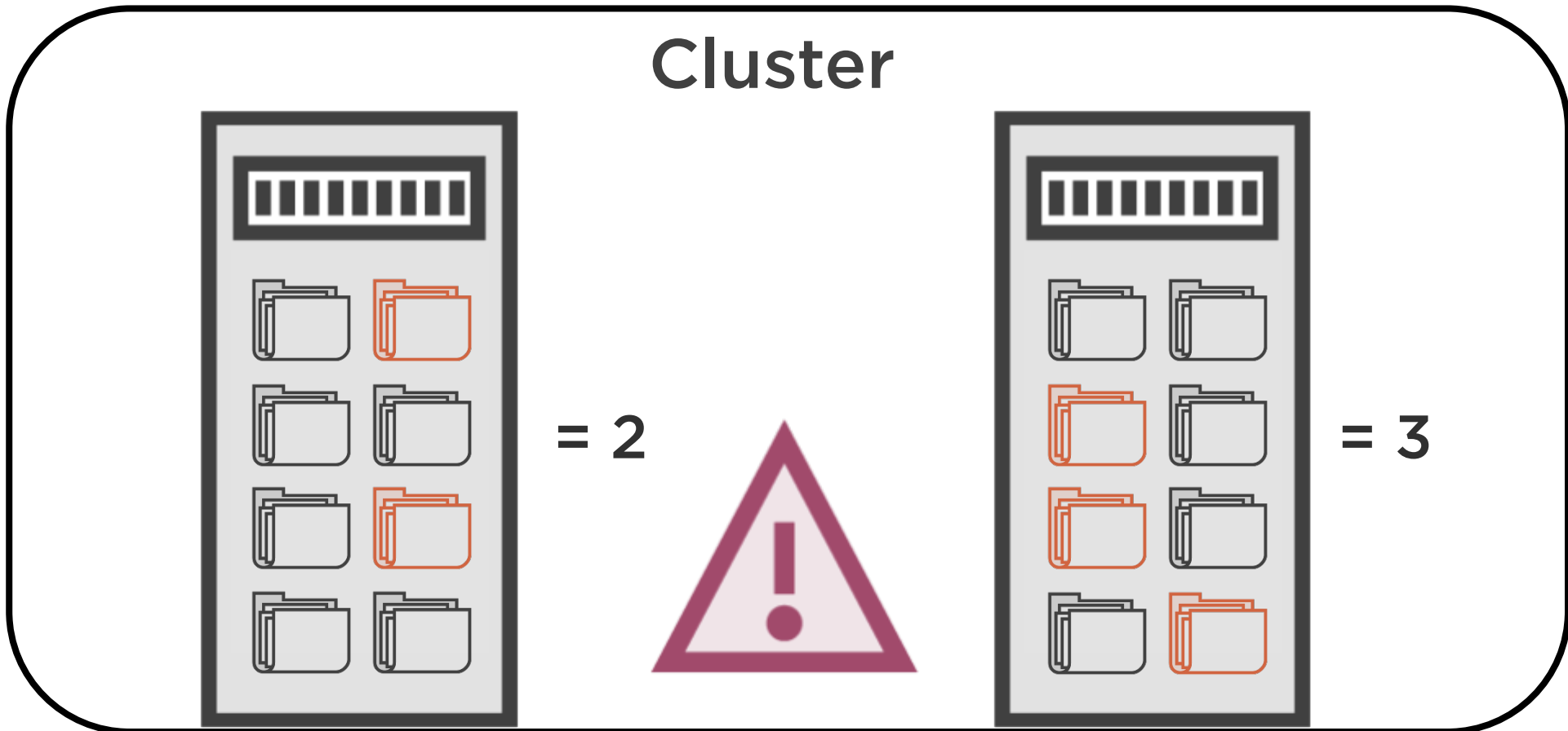


Java Management Extensions



Under-replicated Partitions

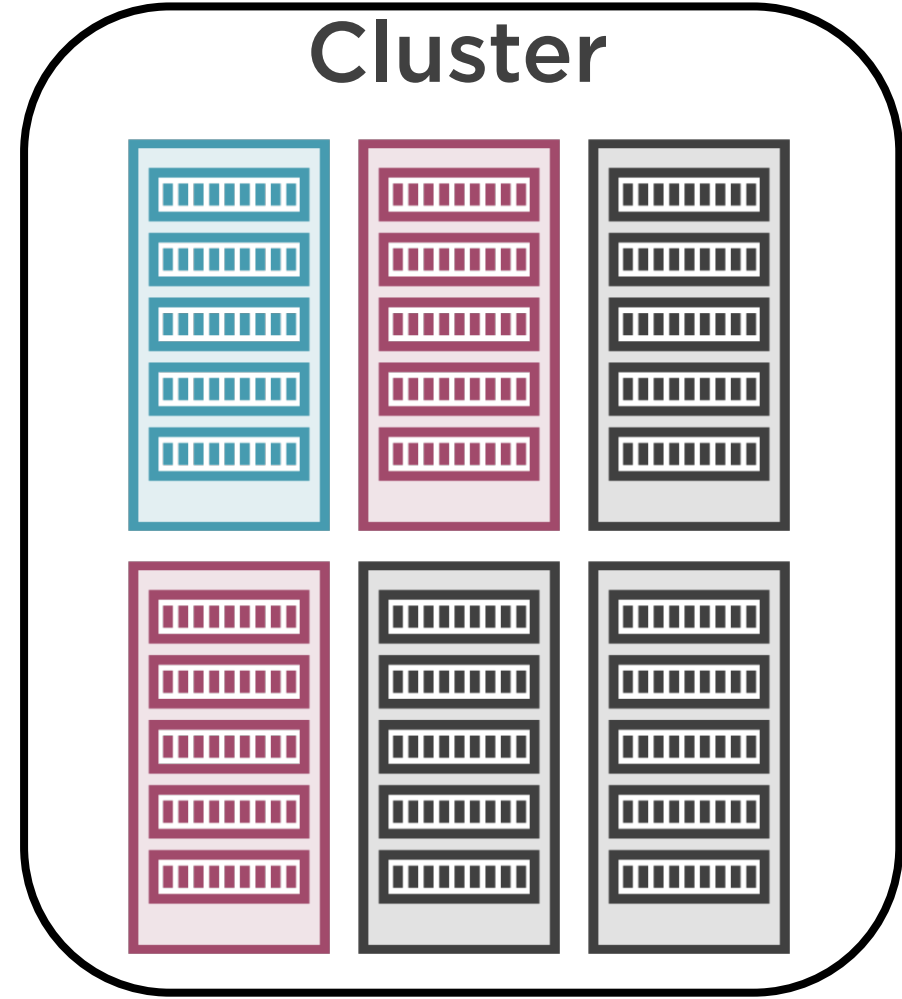
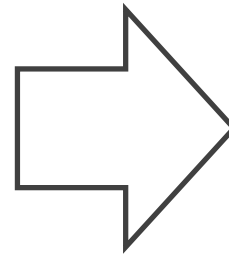
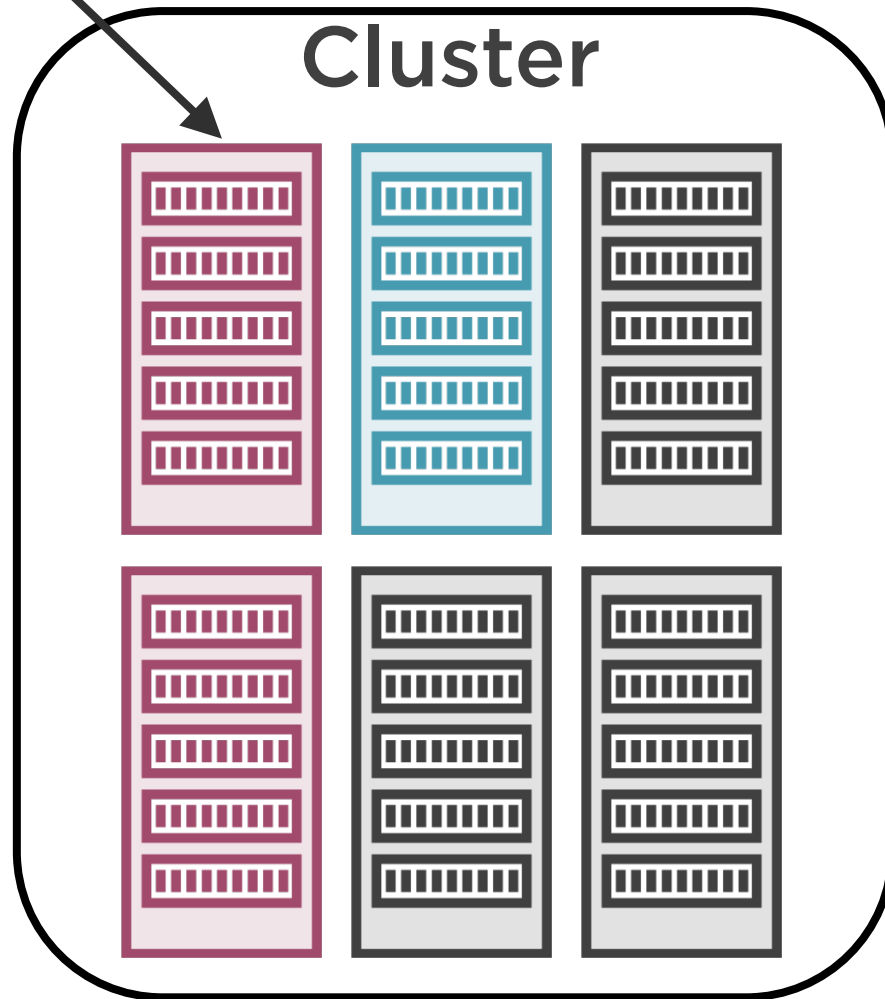
JMX Mbean: `kafka.server:type=ReplicaManager,name=UnderReplicatedPartitions`



Under-replicated Partitions

`auto.leader.rebalance.enable = false`

PREFERRED

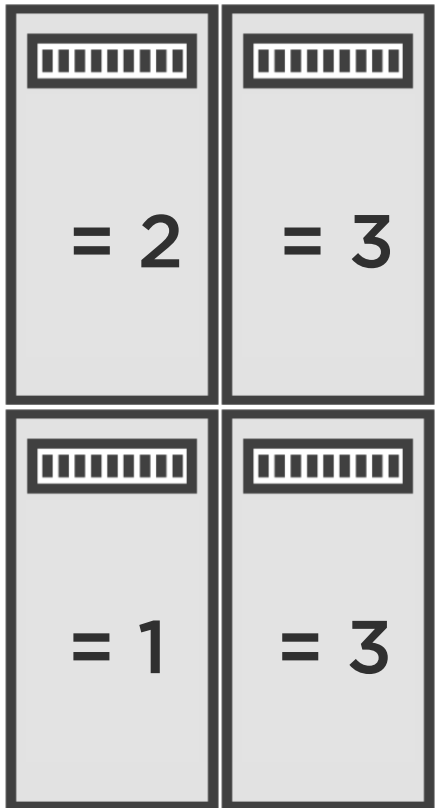


> `kafka-preferred-replica-election`



Under-replicated Partitions

Symptom: Constant $\# > 0$



=



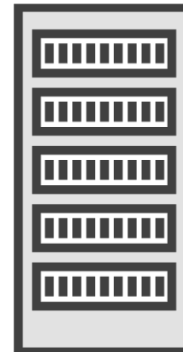
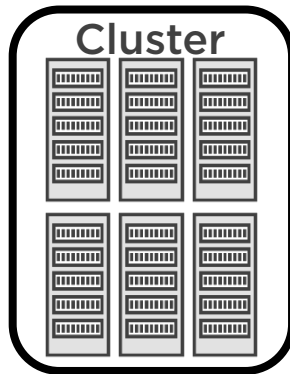
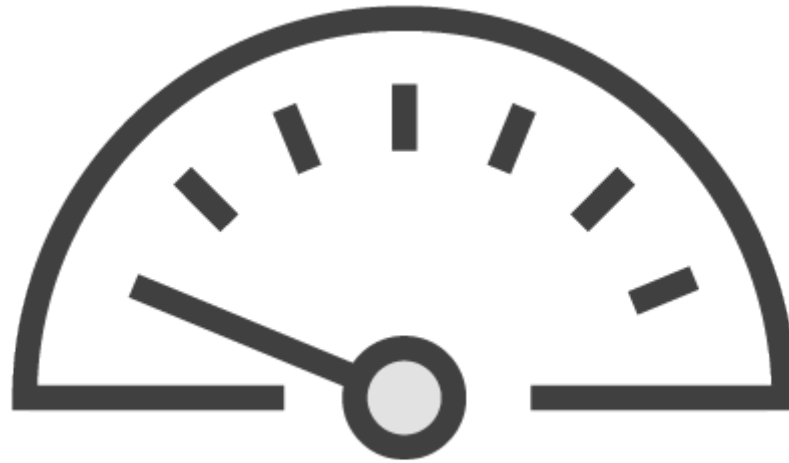
9 partitions



Under-replicated Partitions

Symptom: Changing #

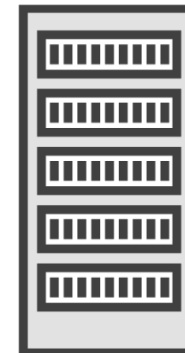
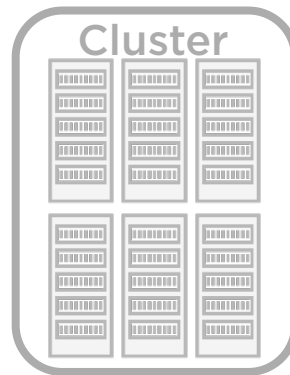
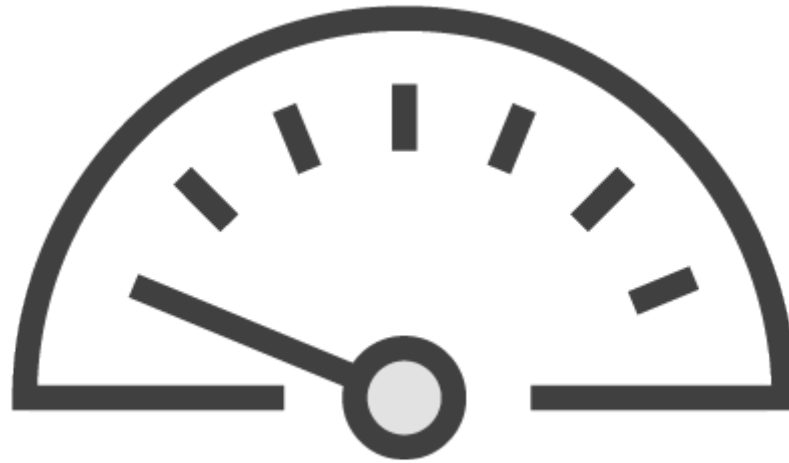
Symptom: Constant # > 0 && No offline broker



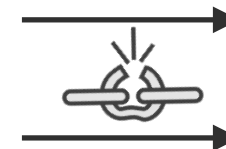
Under-replicated Partitions

Symptom: Changing #

Symptom: Constant # > 0 && No offline broker



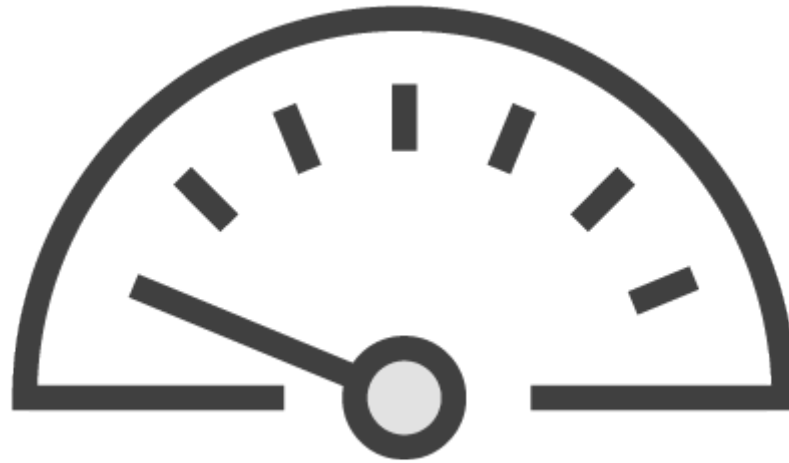
= All under-replicated listed



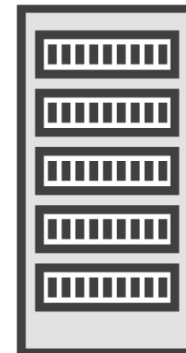
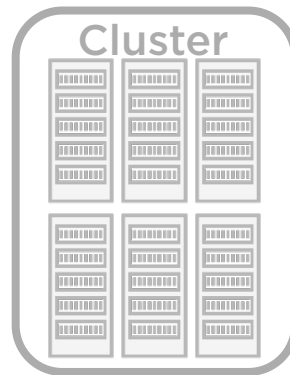
Under-replicated Partitions

Symptom: Changing #

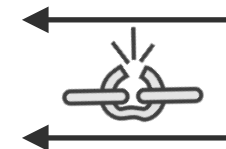
Symptom: Constant # > 0 && No offline broker

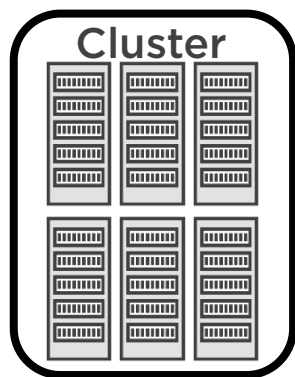
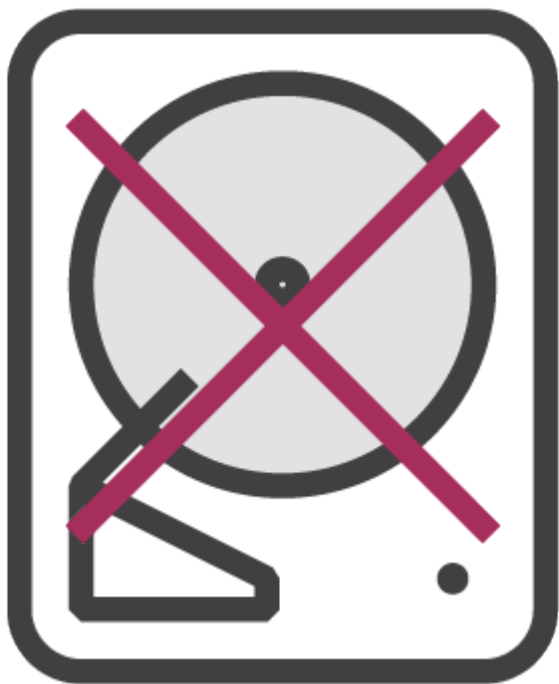


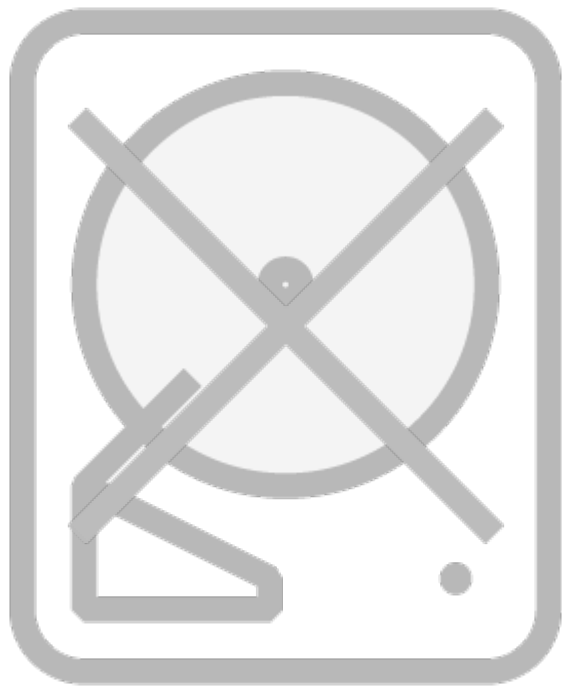
```
> kafka-topics ...  
--describe --under-replicated
```

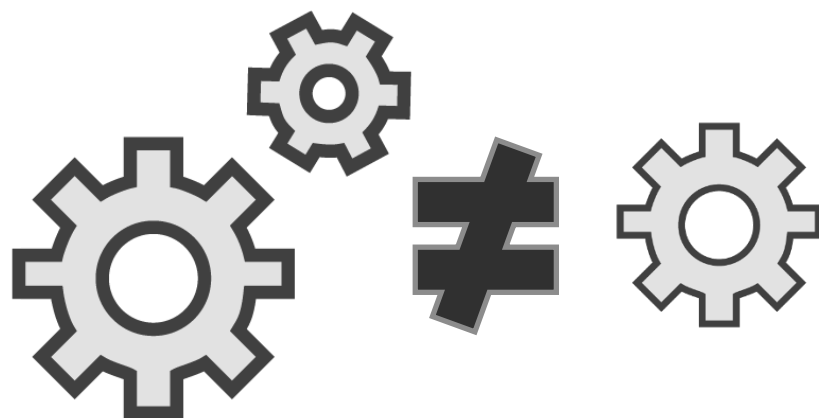


= 0 of under-replicated listed





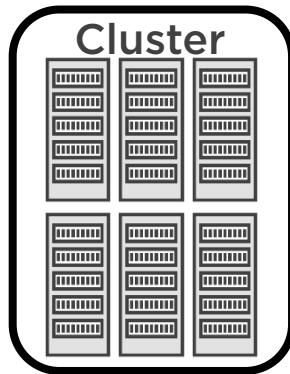




Under-replicated Partitions

Symptom: Changing #

Symptom: Constant # > 0 && No offline broker





Broker Id	Partitions	Leaders	Bytes In	Bytes Out	Messages In
1	40	35	10 MB/s	45 MB/s	425 Msg/s
2	41	35	11 MB/s	48 MB/s	444 Msgs/s
3	40	35	9.8 M	43.3 MB/s	418 Msgs/s
4	39	35	10.1 MB/s	44.7 MB/s	422 Msg/s

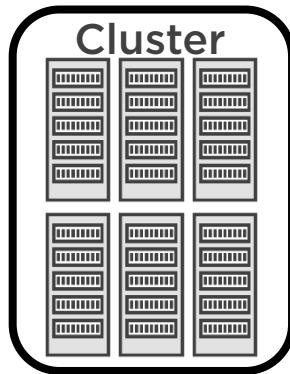
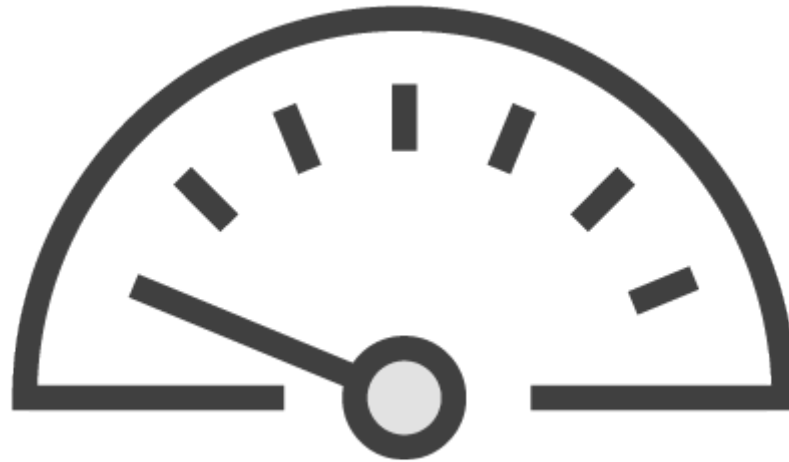


Under-replicated Partitions

Symptom: Changing #

Symptom: Constant # > 0 && No offline broker

```
> kafka-reassign-partitions
```

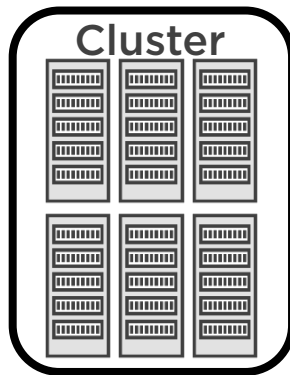


Under-replicated Partitions

Symptom: Changing #

Symptom: Constant # > 0 && No offline broker

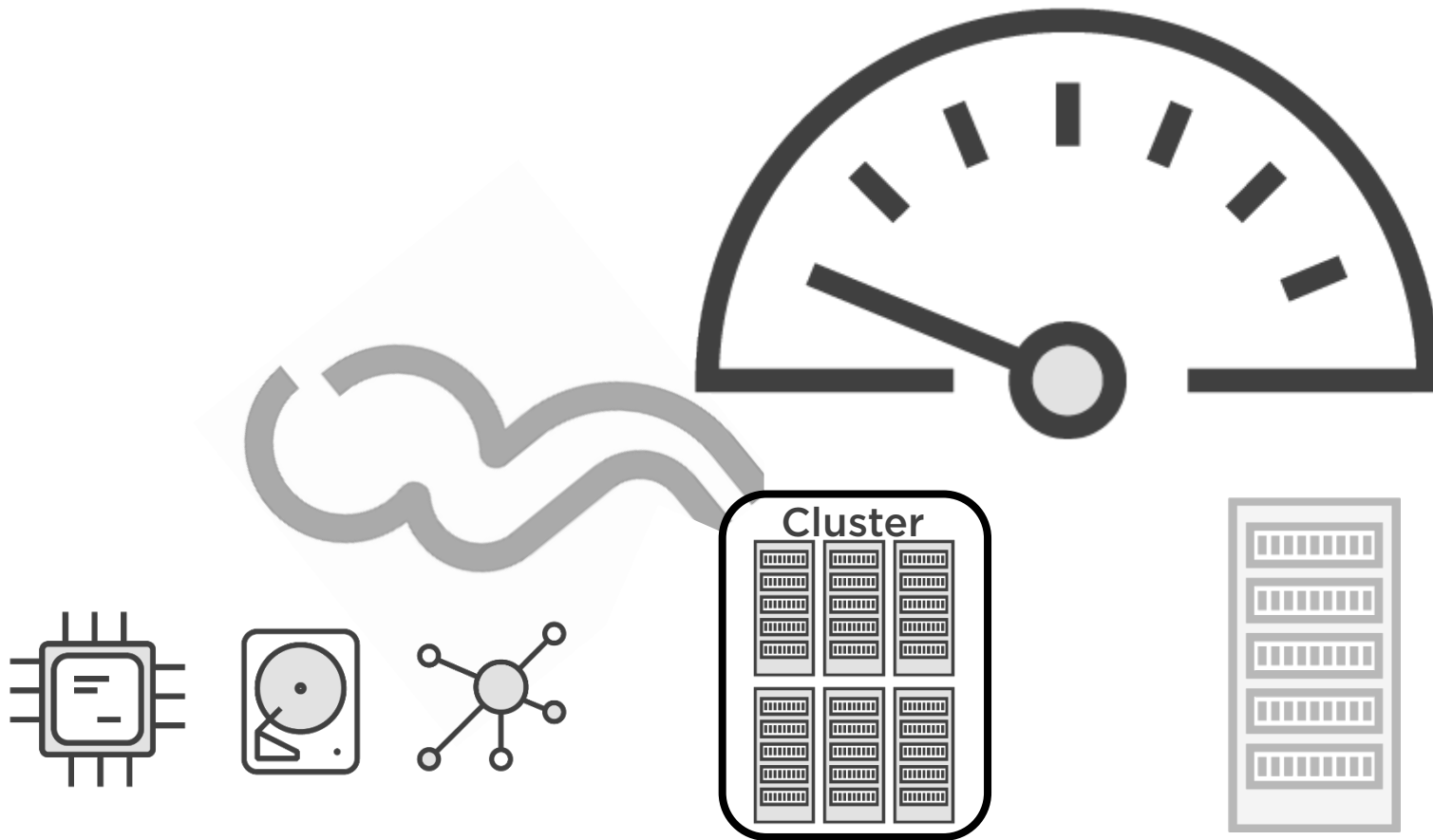
```
> kafka-reassign-partitions
```



Under-replicated Partitions

Symptom: Changing #

Symptom: Constant # > 0 && No offline broker

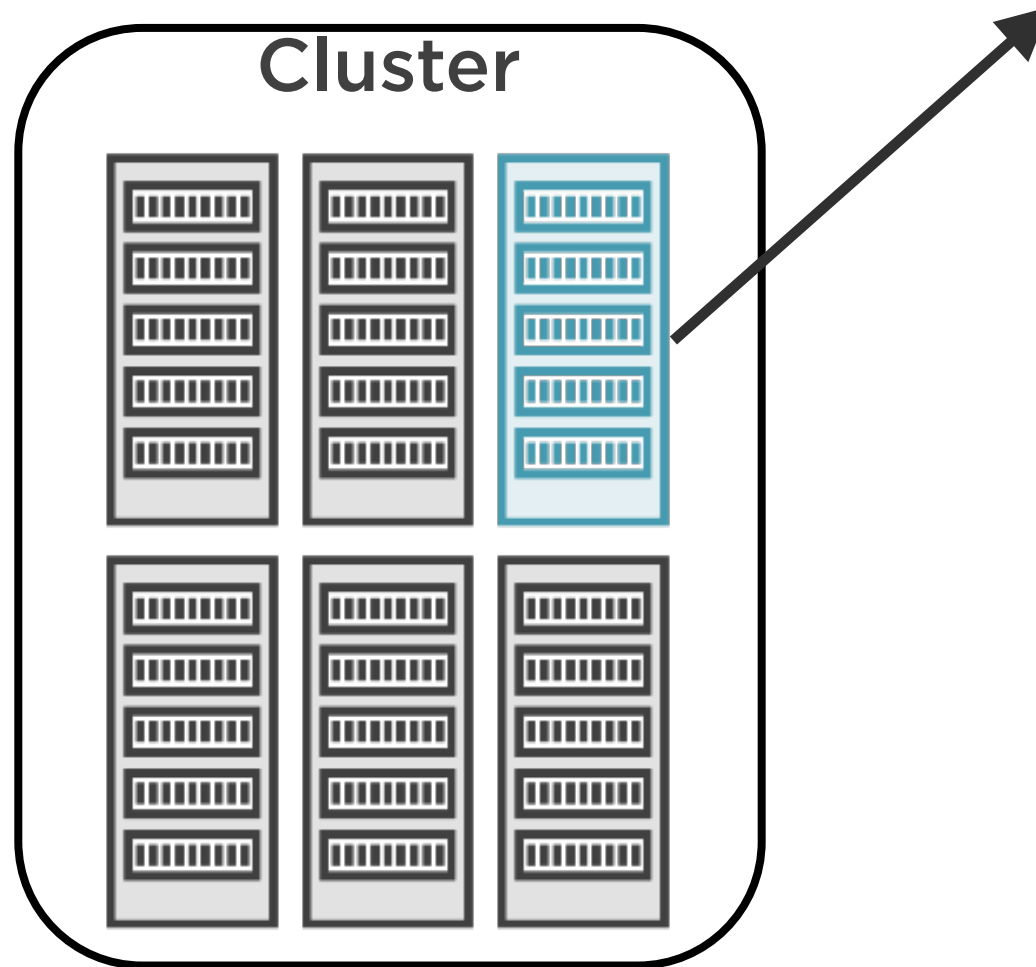


Broker Key Metrics



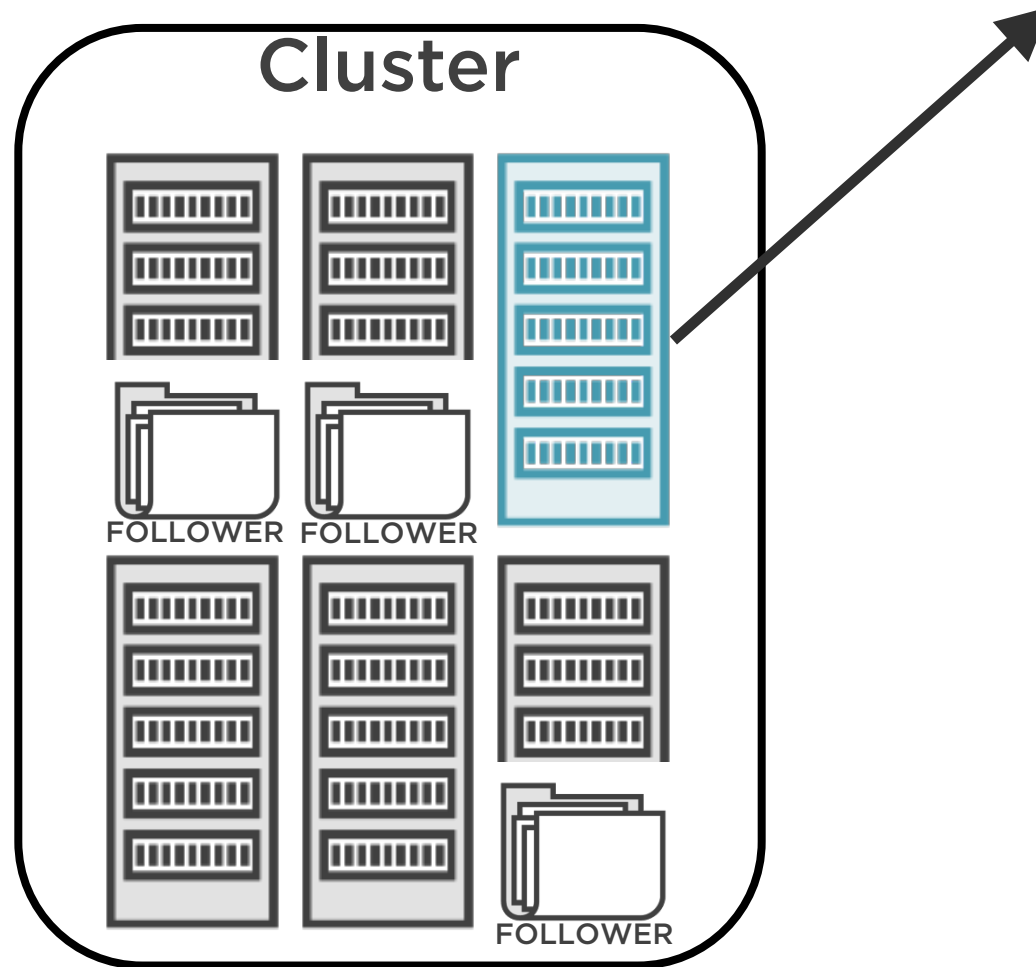
Offline Partitions

JMX Mbean: `kafka.controller:type=KafkaController,name=OfflinePartitionsCount`



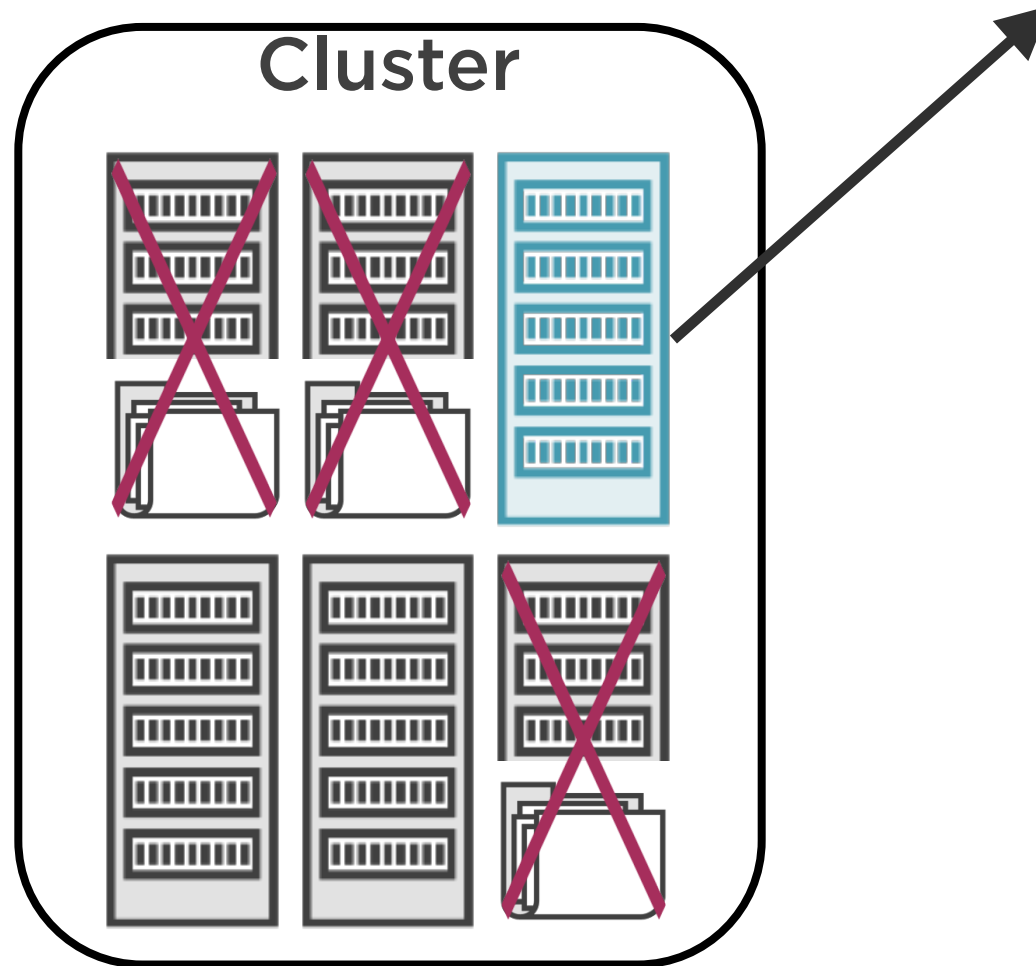
Offline Partitions

JMX Mbean: `kafka.controller:type=KafkaController,name=OfflinePartitionsCount`



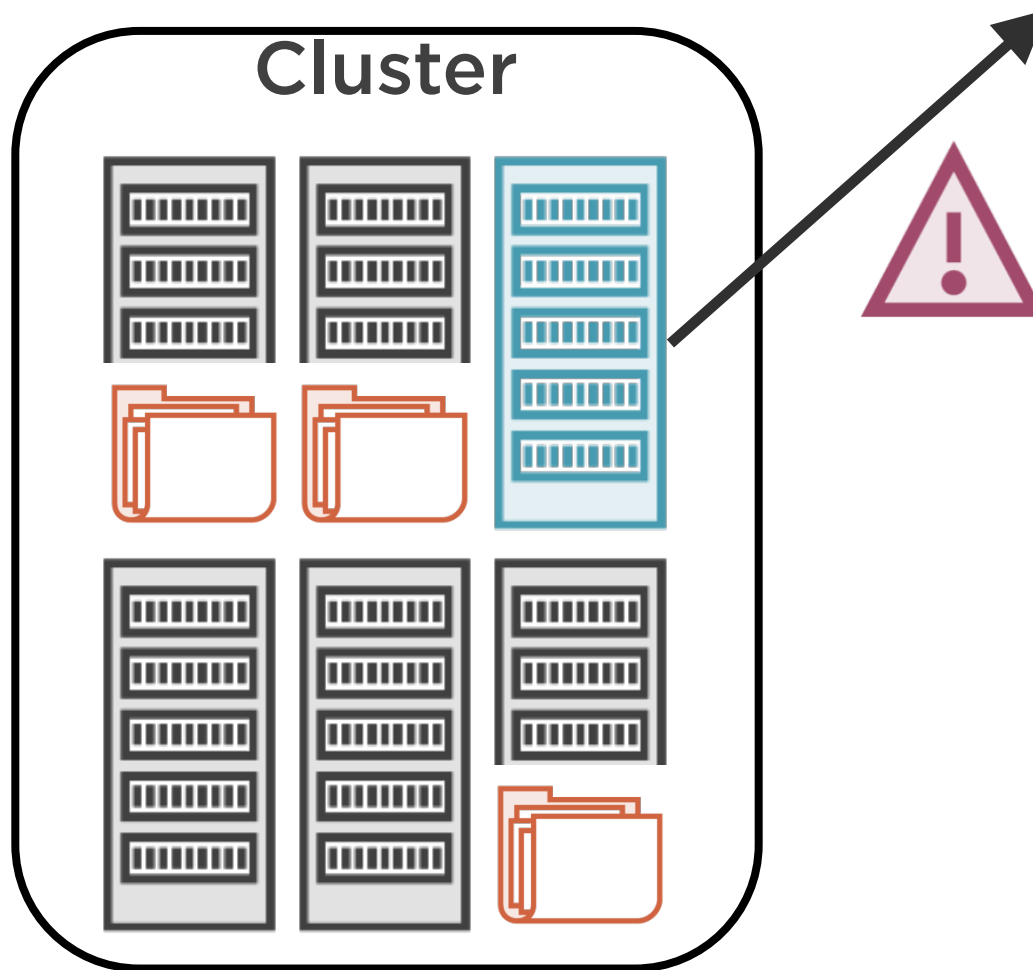
Offline Partitions

JMX Mbean: `kafka.controller:type=KafkaController,name=OfflinePartitionsCount`



Offline Partitions

JMX Mbean: `kafka.controller:type=KafkaController,name=OfflinePartitionsCount`

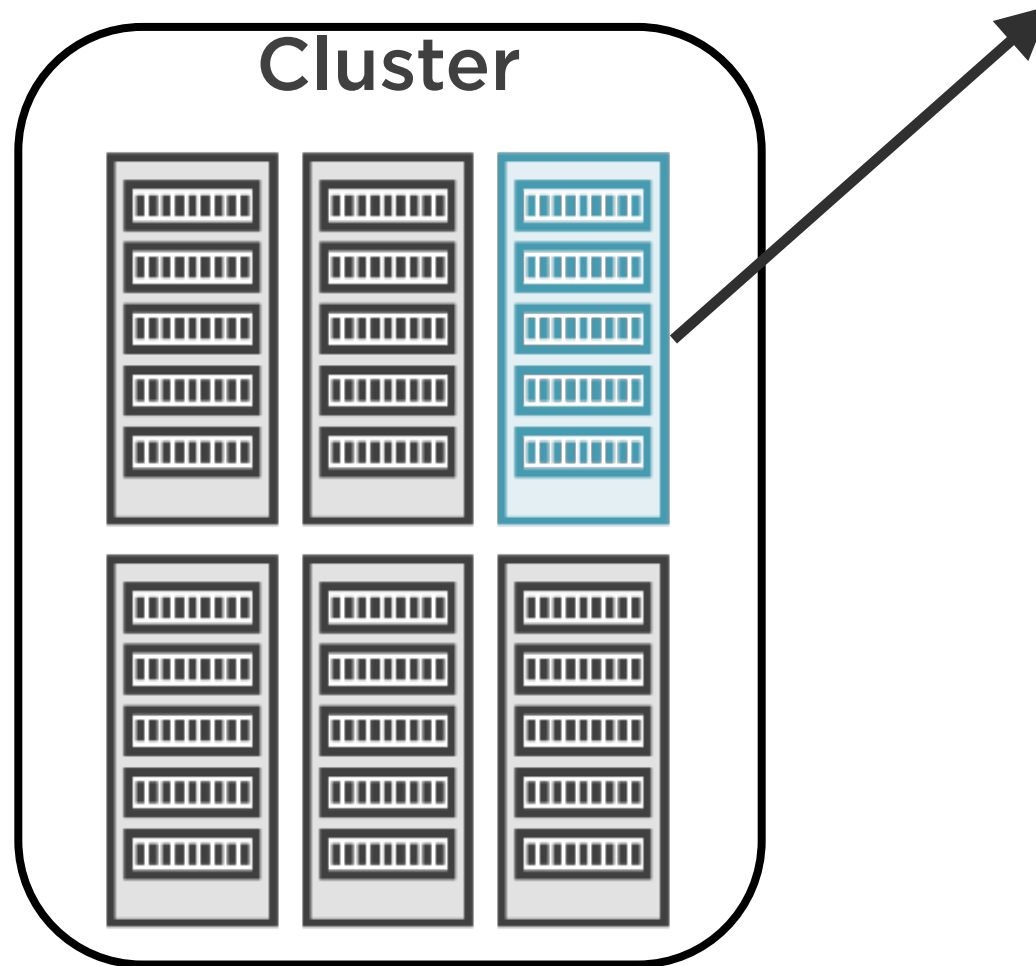


`unclean.leader.election.enable = false`



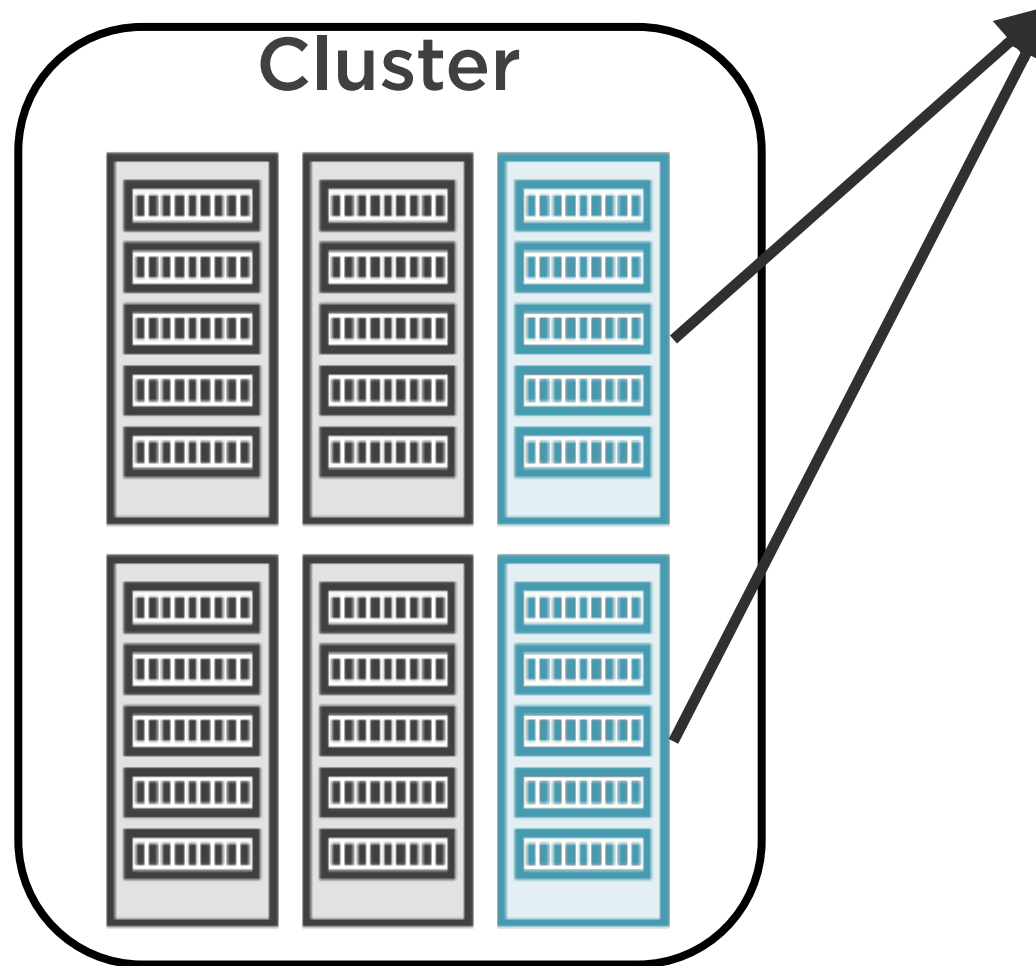
Active Controller Count

JMX Mbean: `kafka.controller:type=KafkaController,name=ActiveControllerCount`



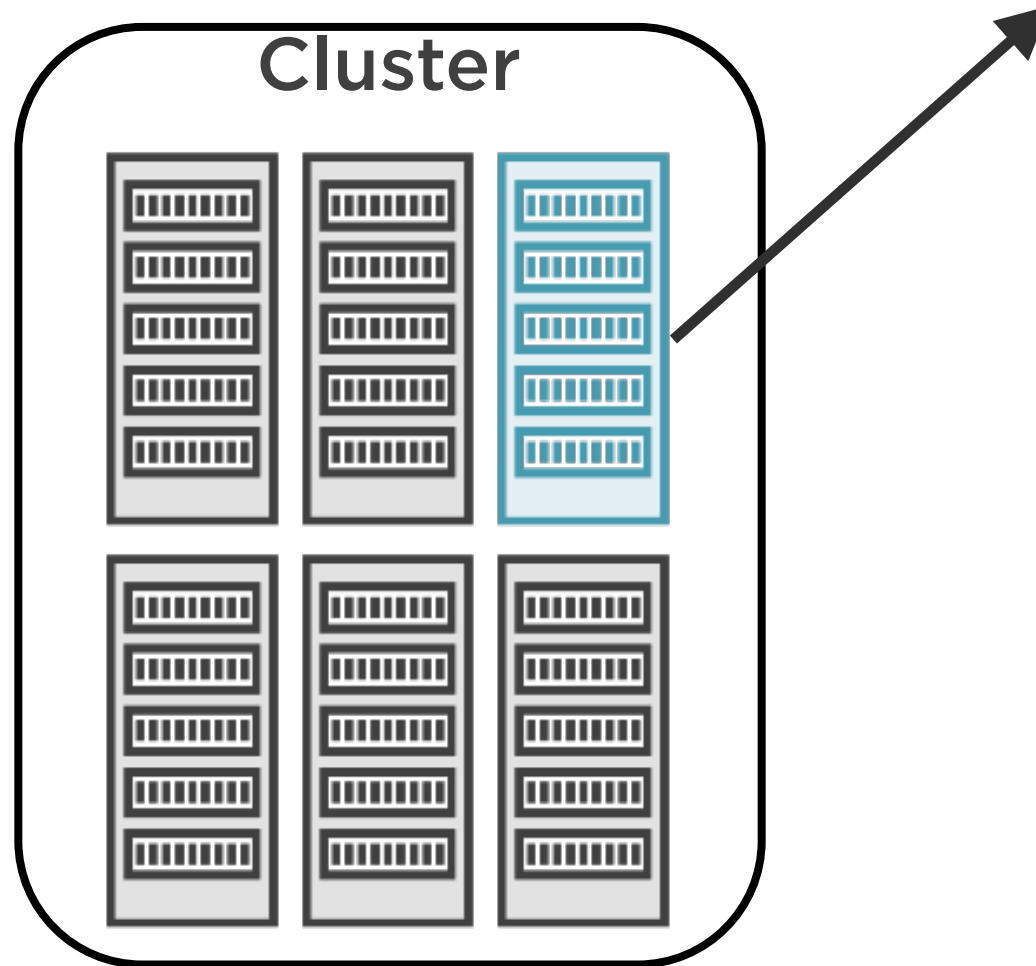
Active Controller Count

JMX Mbean: `kafka.controller:type=KafkaController,name=ActiveControllerCount`



Active Controller Count

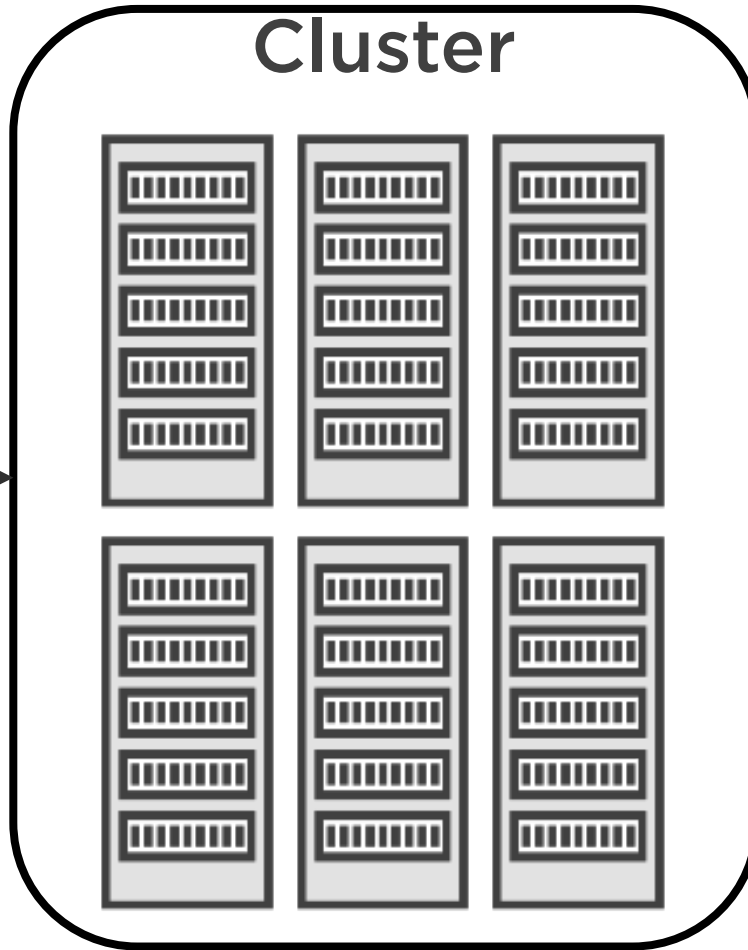
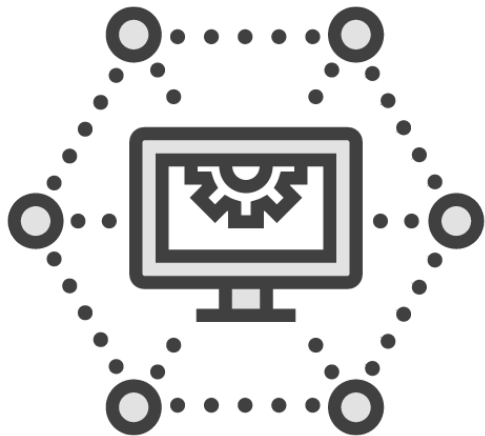
JMX Mbean: `kafka.controller:type=KafkaController,name=ActiveControllerCount`



Active Controller Count

JMX Mbean: `kafka.controller:type=KafkaController,name=ActiveControllerCount`

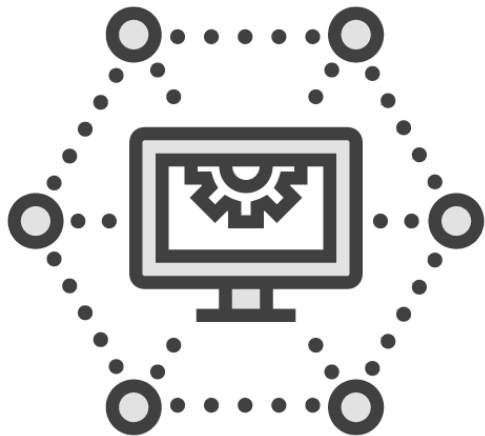
Apache ZooKeeper



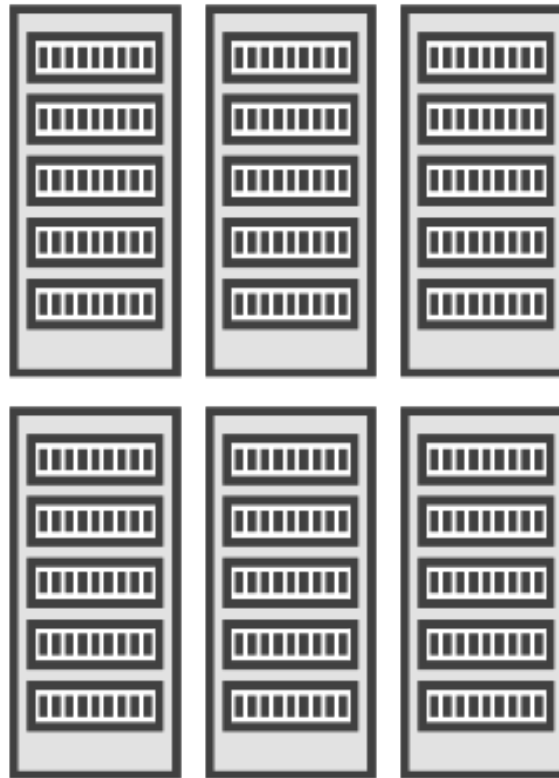
Active Controller Count

JMX Mbean: `kafka.controller:type=KafkaController,name=ActiveControllerCount`

Apache ZooKeeper

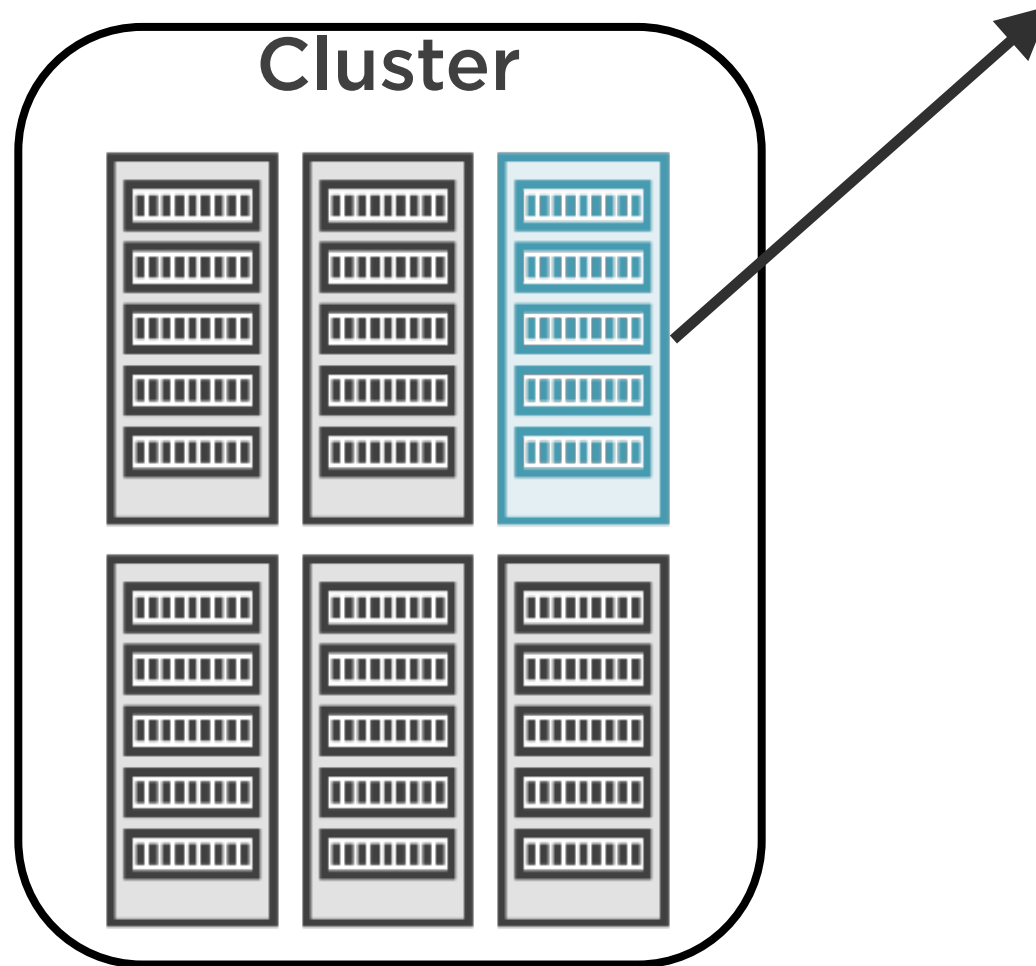


Cluster



Active Controller Count

JMX Mbean: `kafka.controller:type=KafkaController,name=ActiveControllerCount`

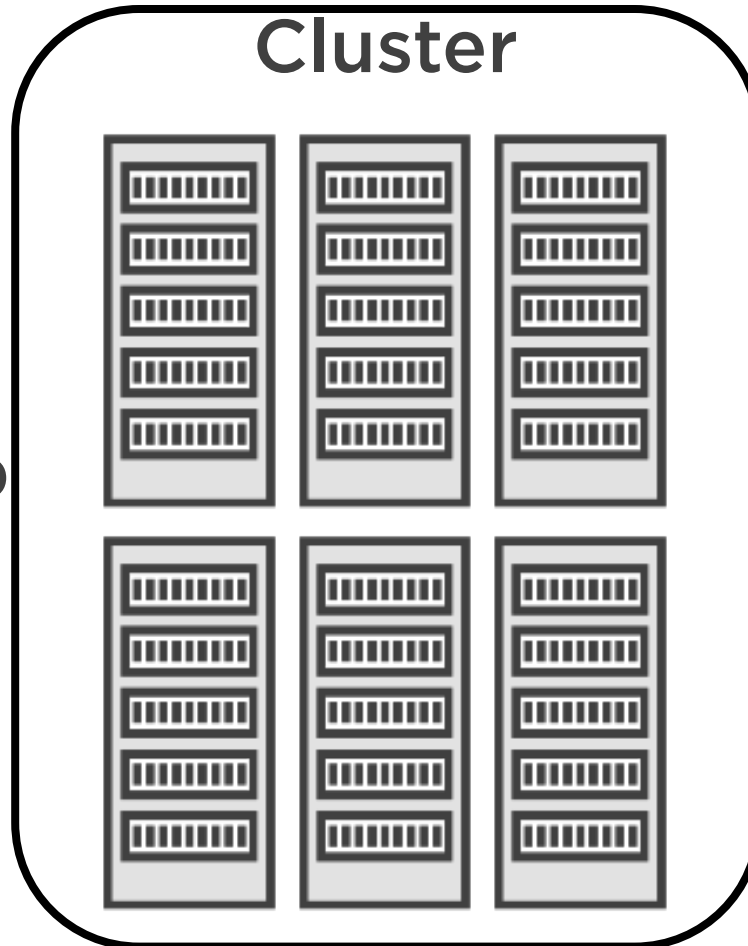


All Topic Bytes In/Out

JMX Mbean: kafka.server:type=BrokerTopicMetrics,name=BytesInPerSec

JMX Mbean: kafka.server:type=BrokerTopicMetrics,name=BytesOutPerSec

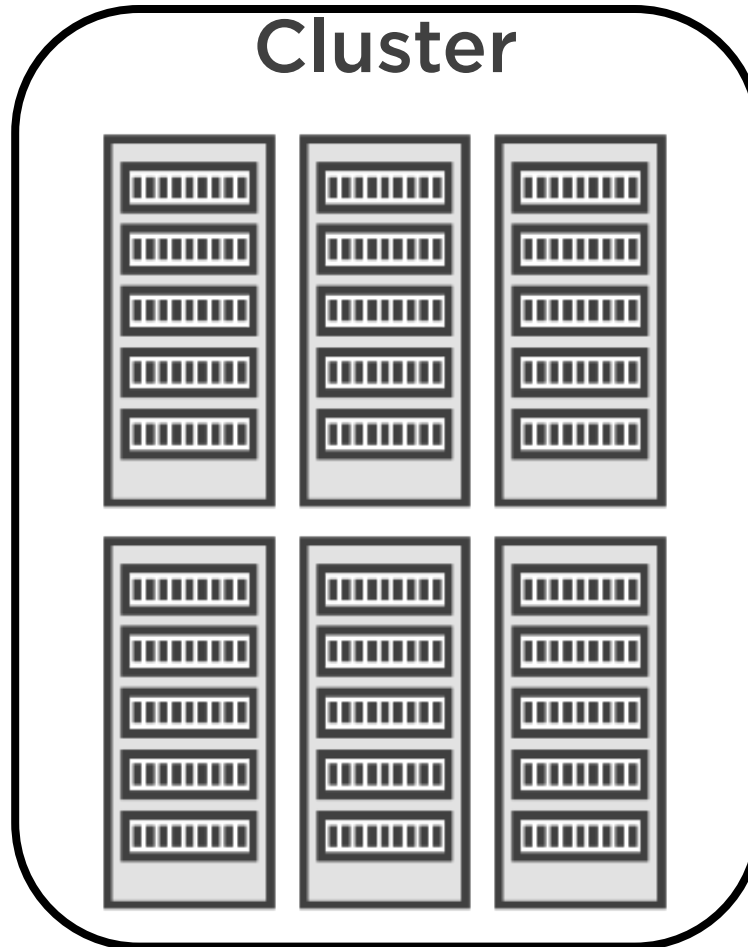
10101110101000101010



All Topic Bytes In/Out

JMX Mbean: kafka.server:type=BrokerTopicMetrics,name=BytesInPerSec

JMX Mbean: kafka.server:type=BrokerTopicMetrics,name=BytesOutPerSec



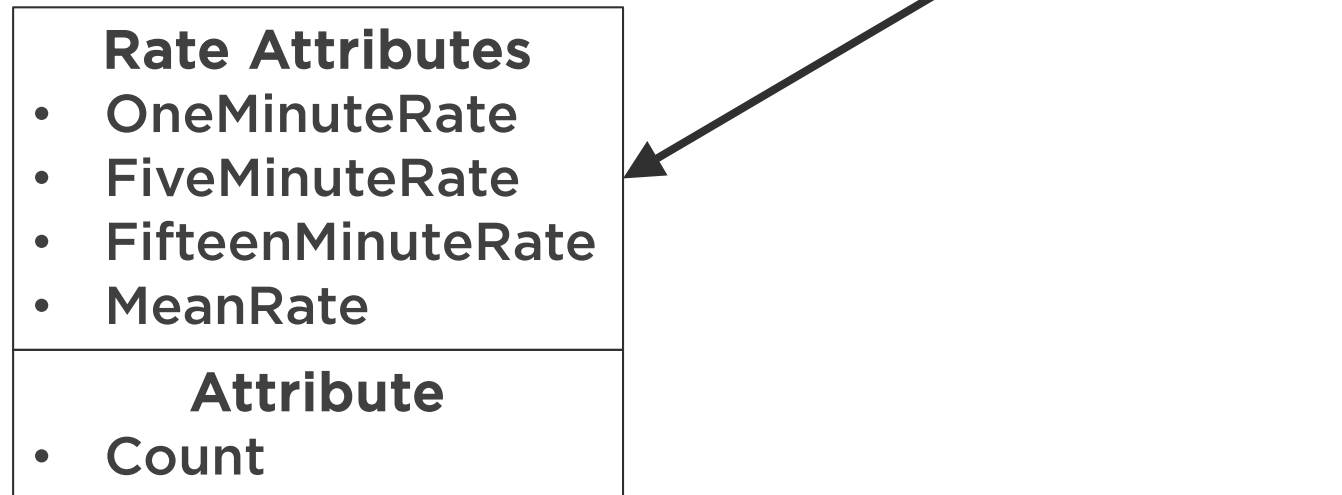
0 1 1 0 1 0 1 0 1 1 1 0 1 0 1 0 0 0 1 0
1 0 0 1 1 1 0 0 1 1 1 0 0 1
1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1
1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1



All Topic Bytes In/Out

JMX Mbean: kafka.server:type=BrokerTopicMetrics,name=BytesInPerSec [ATTRIBUTES]

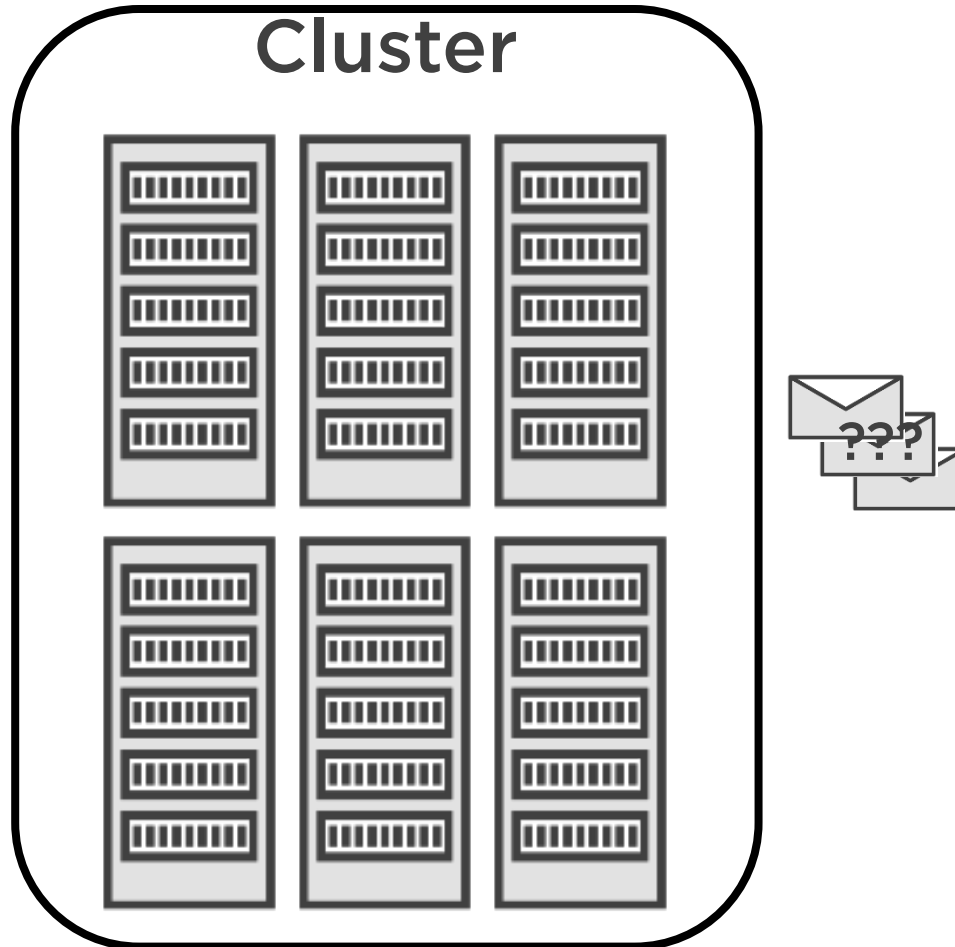
JMX Mbean: kafka.server:type=BrokerTopicMetrics,name=BytesOutPerSec [ATTRIBUTES]



Rate Attributes <ul style="list-style-type: none">• OneMinuteRate• FiveMinuteRate• FifteenMinuteRate• MeanRate
Attribute <ul style="list-style-type: none">• Count

All Topics Messages In

JMX Mbean: `kafka.server:type=BrokerTopicMetrics,name=MessagesInPerSec`



Partition Count

JMX Mbean: kafka.server:type=ReplicaManager,name=PartitionCount

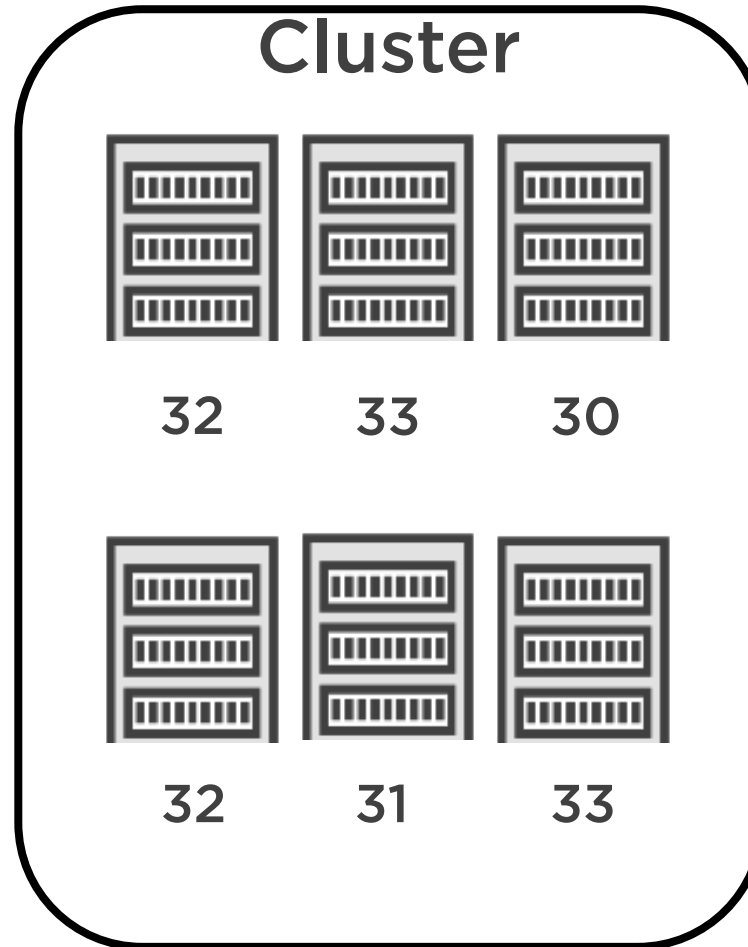
```
auto.create.topics.enable = true
```



Leader Count

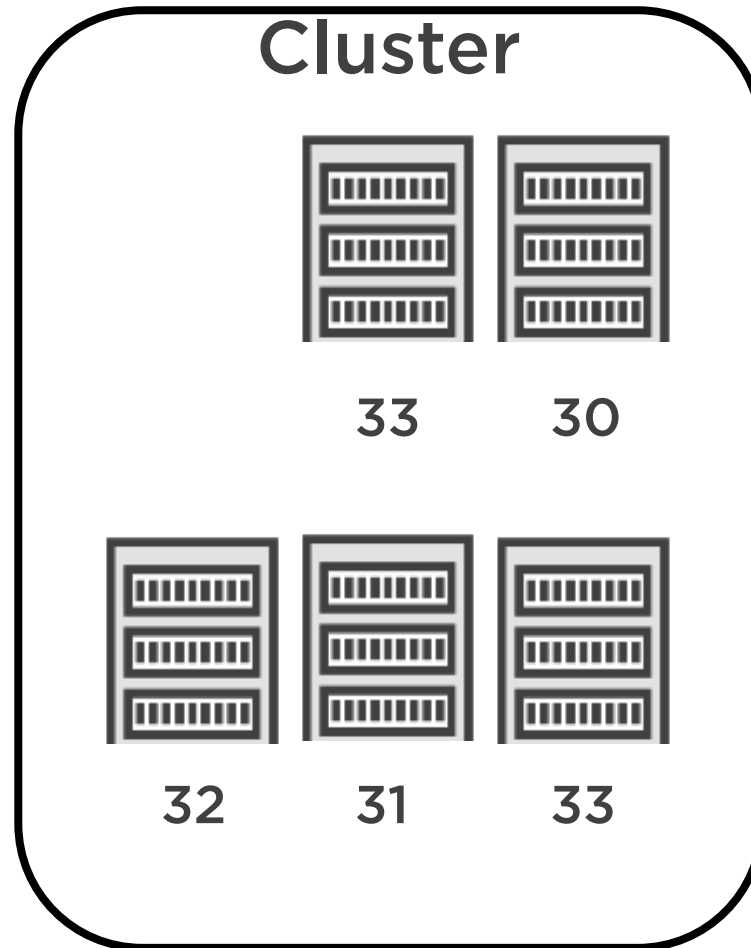
JMX Mbean: `kafka.server:type=ReplicaManager,name=LeaderCount`

`auto.leader.rebalance.enable = false`



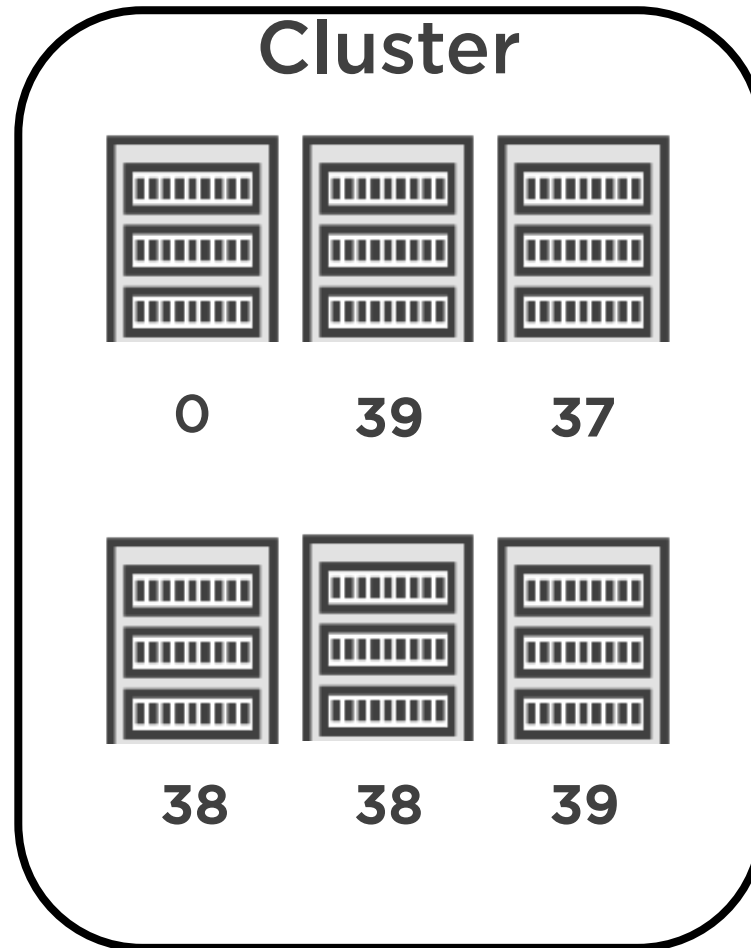
Leader Count

JMX Mbean: `kafka.server:type=ReplicaManager,name=LeaderCount`



Leader Count

JMX Mbean: kafka.server:type=ReplicaManager,name=LeaderCount

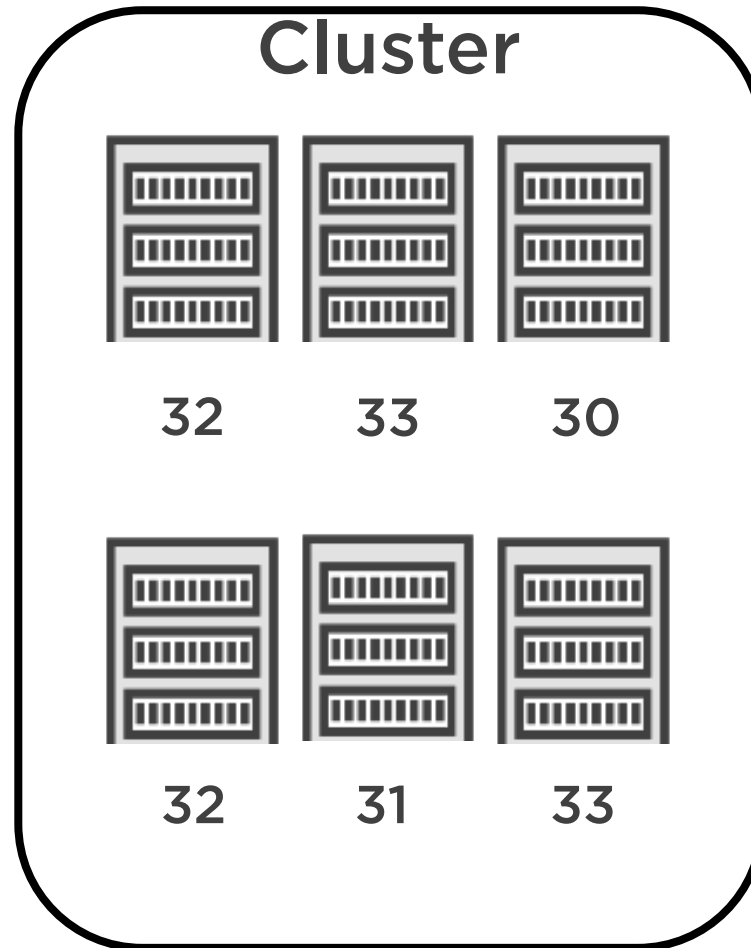


```
> kafka-preferred-replica-election
```



Leader Count

JMX Mbean: kafka.server:type=ReplicaManager,name=LeaderCount



Leader Count

JMX Mbean: kafka.server:type=ReplicaManager,name=LeaderCount

$$\frac{\text{LeaderCount}}{\text{PartitionCount}} = \frac{1}{\text{ReplicationFactor}}$$



Request Metrics

JMX Mbean: kafka.network:type=RequestMetrics,name=[METRIC_TYPE],request=[REQUEST_NAME]

Metric Type

- TotalTimMs
- RequestQueueTimeMs
- LocalTimeMs
- RemoteTimeMs
- ThrottleTimeMs
- ResponseQueueTimeMs
- ResponseSendTimeMs
- RequestsPerSec

Request Name

- ApiVersions
- ControlledShutdown
- CreateTopics
- DeleteTopics
- DescribeGroups
- ...
- Produce
- SaslHandshake
- StopReplica
- SyncGroup
- UpdateMetadata

Attributes

- 50thPercentile
- 75thPercentile
- ..
- 99thPercentile
- 999thPercentile
- Count
- Min
- Max
- Mean
- StdDev



Partition and Topic Metrics

JMX Mbean: ...,topic=[TOPIC_NAME],partition=[PARTITION_NUMBER]



Client Key Metrics



Producer Metrics

JMX Mbean: kafka.producer:type=producer-metrics,client-id=[CLIENT_ID]

JMX Mbean: kafka.producer:type=producer-**node**-metrics,client-id=[CLIENT_ID]

,**node-id=node-[BROKER_ID]**

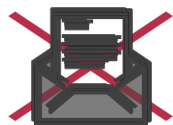
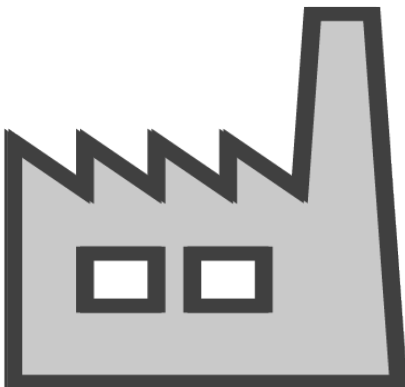
JMX Mbean: kafka.producer:type=producer-**topic**-metrics,client-id=[CLIENT_ID]

,**topic=[TOPICNAME]**

client.id

Attributes

- record-error-rate
- request-latency-avg
- outgoing-byte-rate
- record-send-rate
- request-rate
- record-queue-time-avg



Consumer Metrics

JMX Mbean: kafka.producer:type=consumer-metrics,client-id=[CLIENT_ID]

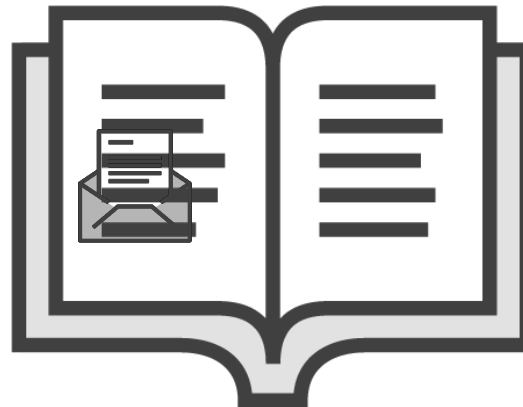
JMX Mbean: kafka.producer:type=consumer-**node**-metrics,client-id=[CLIENT_ID]
 ,node-id=node-[BROKER_ID]

Consumer Metrics

JMX Mbean: kafka.producer:type=consumer-fetch-manager-metrics,client-id=[CLIENT_ID]
optional → ,topic=[TOPICNAME]

Attributes

- fetch-latency-avg
- bytes-consumed-rate
- records-consumed-rate



Consumer Metrics

JMX Mbean: `kafka.producer:type=consumer-coordinator-metrics,client-id=[CLIENT_ID]`

Attributes

- `sync-time-avg`
- `sync-rate`
- `commit-latency-avg`
- `assigned-partitions` metric

Consumer Group



Consumer Metrics Lag Monitoring

github.com/linkedin/Burrow



Quota Metrics

Consumer

JMX Mbean: kafka.consumer:type=consumer-fetch-manager-metrics,client-id=[CLIENT_ID]

Attributes

- fetch-throttle-time-avg

Producer

JMX Mbean: kafka.consumer:type=producer-metrics,client-id=[CLIENT_ID]

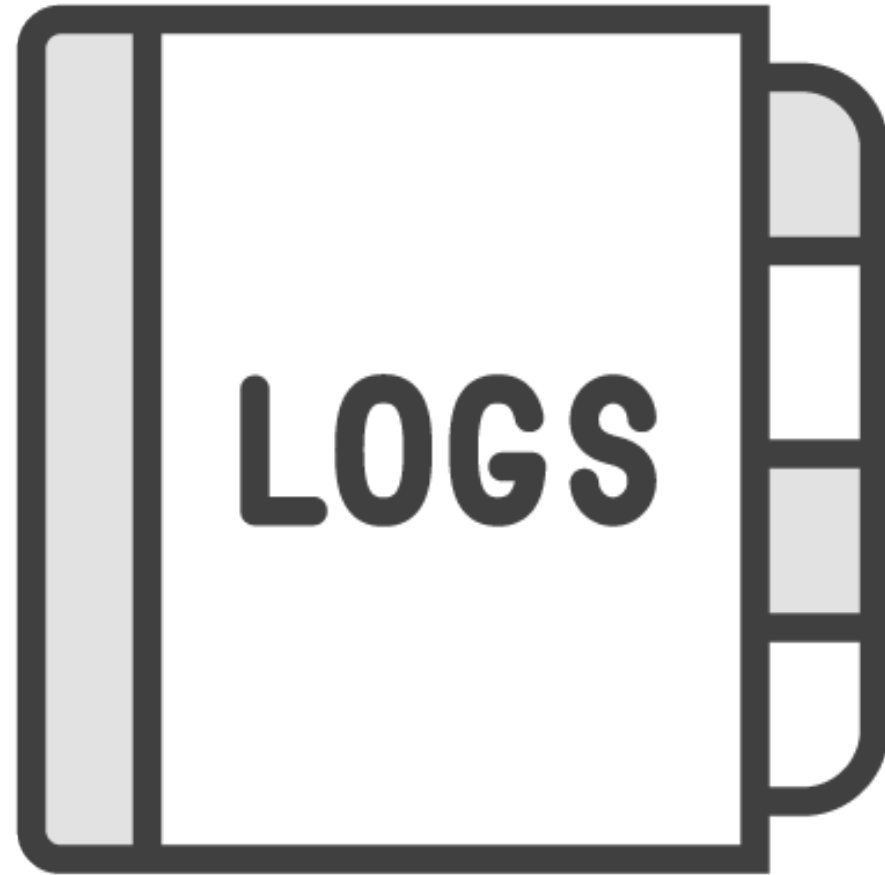
Attributes

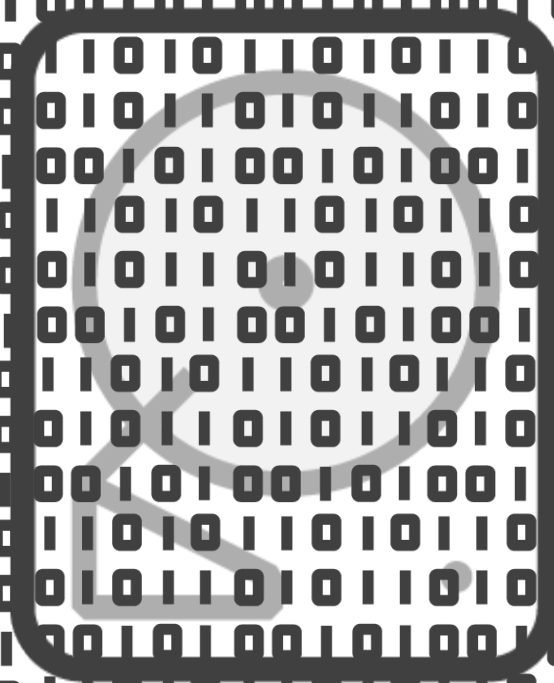
- produce-throttle-time-avg

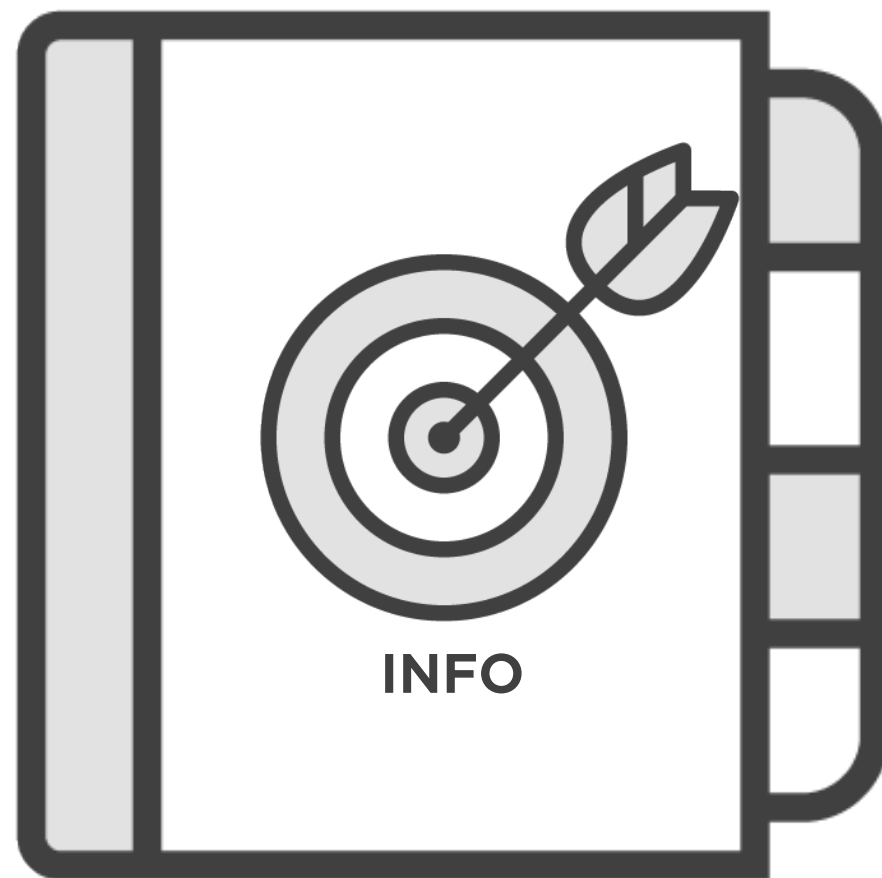


Maintenance Beyond Kafka Metrics











Create Topic 'A'
2 -> 3 Brokers
Partition B-1 Moved
...



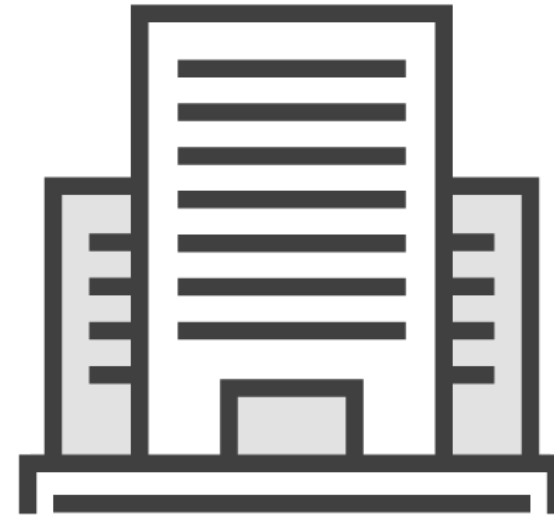
Producer 'x' throttled
Producer 'y' upper bound
Client 'z' exempt request
...



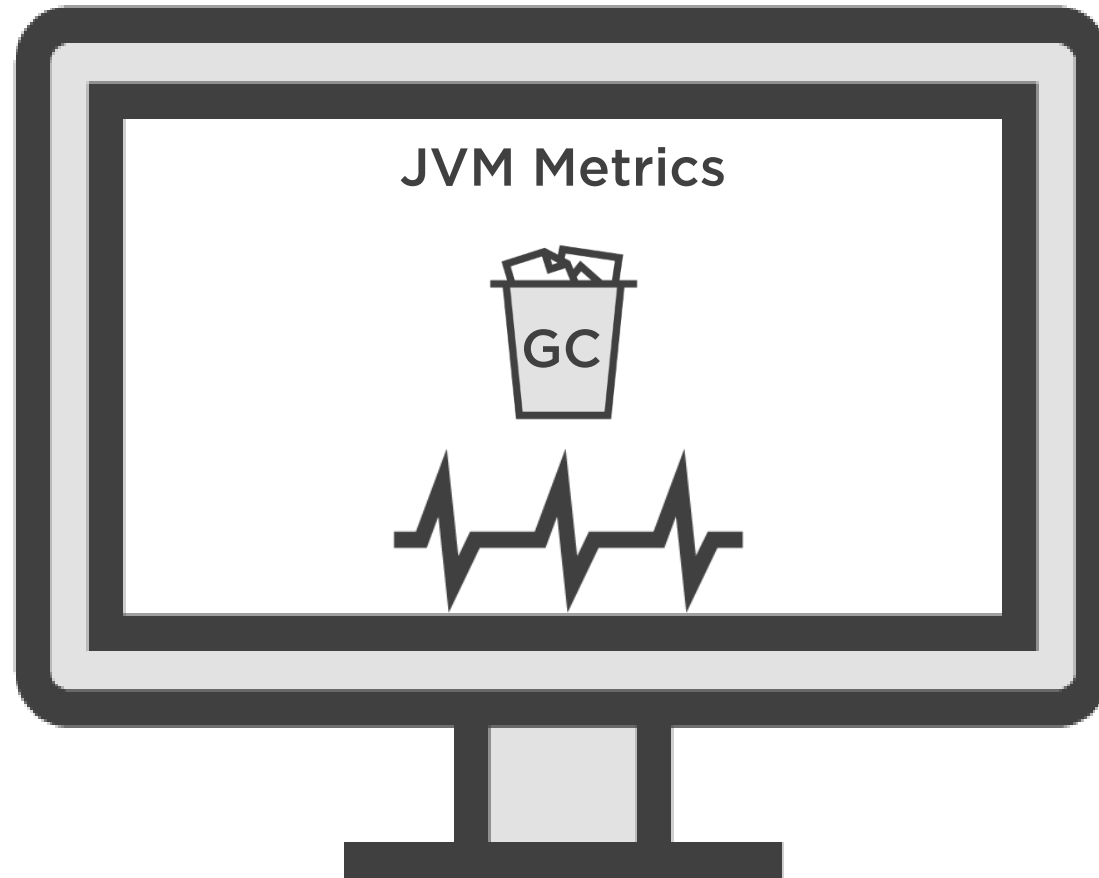




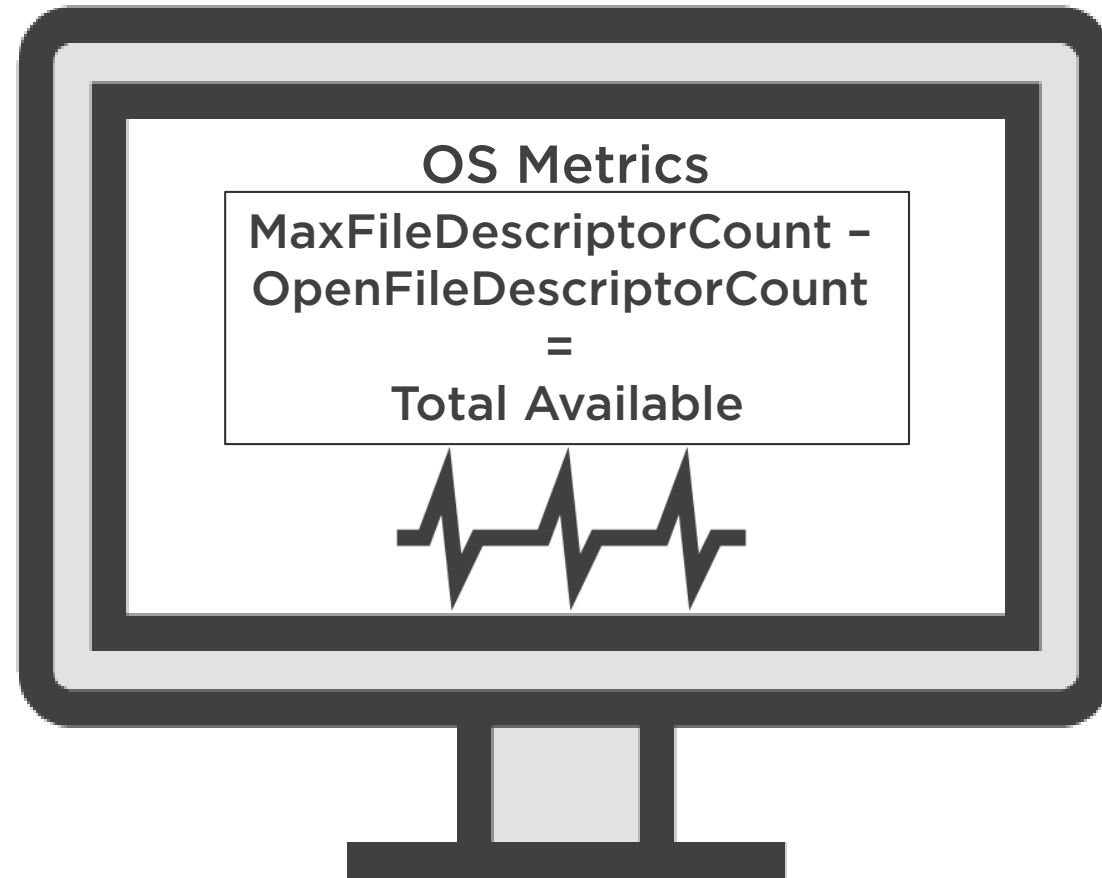
Testing



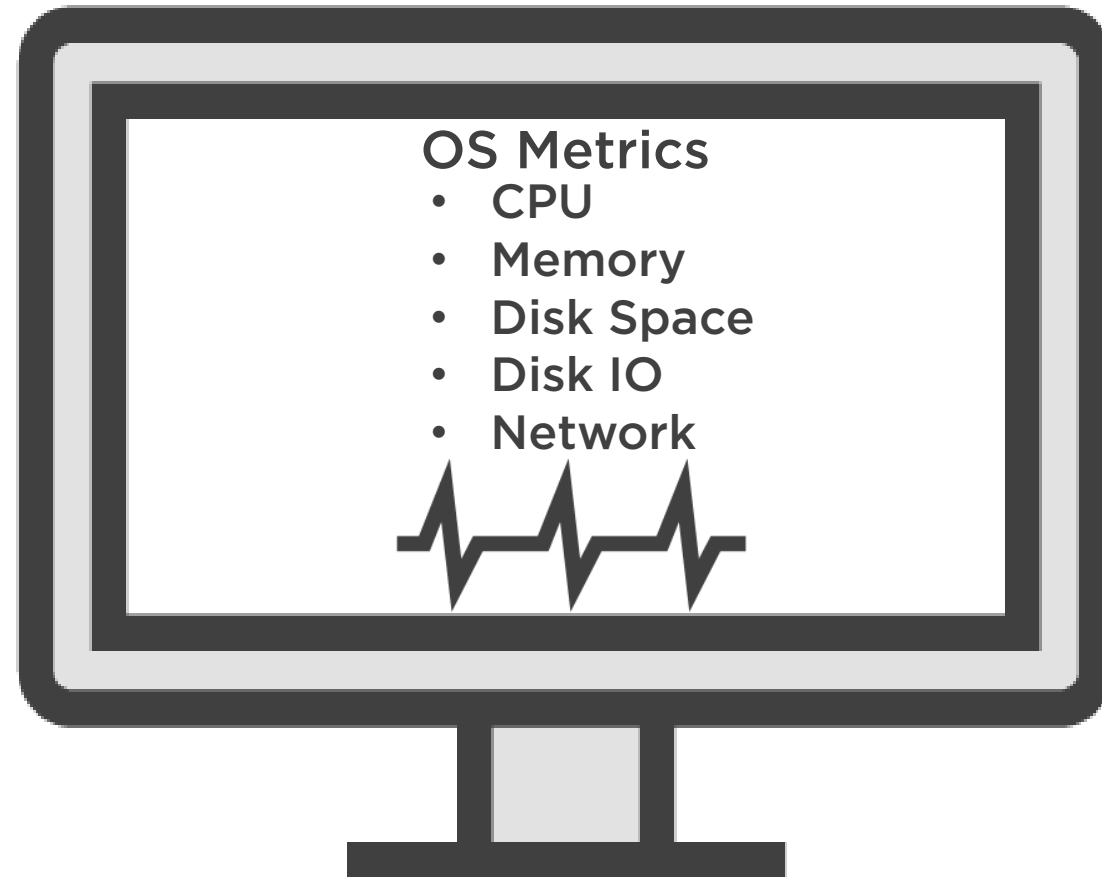
Monitoring – Again!



Monitoring – Again!



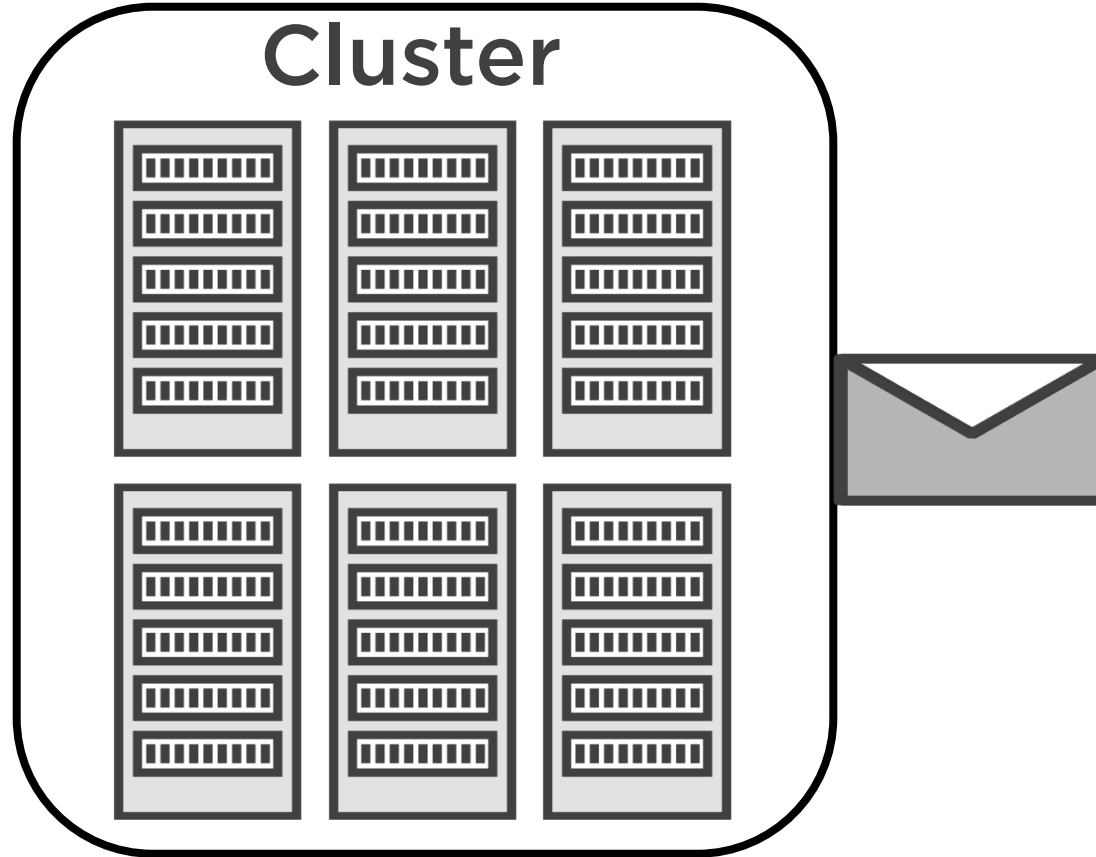
Monitoring – Again!



Monitoring – Again!



Monitoring – Again!



Resources

- **Kafka Documentation: Metrics**
 - kafka.apache.org/documentation/#monitoring
- **Confluent Documentation: Metrics**
 - <https://docs.confluent.io/current/kafka/monitoring.html>
- **Kafka Quotas**
 - kafka.apache.org/documentation/#quotas
- **Monitoring Kafka in Production**
 - logz.io/blog/monitoring-kafka-in-production
- **Monitoring Kafka with Kafka exporter + Prometheus + Grafana**
 - medium.com/@danielmrosa/monitoring-kafka-b97d2d5a5434
- **Burrow: Kafka Consumer Monitoring Reinvented**
 - engineering.linkedin.com/apache-kafka/burrow-kafka-consumer-monitoring-reinvented
- **Introducing Kafka Cruise Control Frontend**
 - engineering.linkedin.com/blog/2019/02/introducing-kafka-cruise-control-frontend



Summary



Next?

Monitoring Basics

Key Metrics

Monitoring in Action

Parting Maintenance Notes

